

UNIVERSIDAD DE CÁDIZ

FACULTAD DE CIENCIAS

MÁSTER EN MATEMÁTICAS

MODELIZACIÓN MATEMÁTICA Y ANÁLISIS DE DATOS EN LEUCEMIA

Trabajo de fin de máster presentado por

Ana del Rosario Niño López

Tutora: Dra. María Rosa Durán

Tutor: D. Álvaro Martínez Rubio

Puerto Real, 9 de diciembre de 2020

Abstract

Mathematical Oncology is a quite recent discipline which is expected to pin down solutions to problems related to cancer and to take advantage of all available mathematical tools.

Specifically, Acute Lymphoblastic Leukaemia (ALL), which concerns blood cells, will be examined. ALL brings about 80% of leukaemias when coming down to pediatric ages. Survival of these patients has increased by a considerable amount in recent years, however, around 15 – 20% of treatments are unsuccessful. It implies a relapse. For this reason, it is definitely required to come up with new strategies to study and select which patients are at higher risk of relapse.

This assignment will develop mathematical models which describe the behavior of the disease, examining the evolution of a leukaemic clone on the basis of real data. Therefore, it is suggested a model which allows testing future alternative therapies. Methodology employed will combine ordinary differential equations, numerical simulations, data analysis techniques along with statistical tools with data processing with *Python* and *Mathematica*.

A mi familia.

Resumen

La Oncología Matemática es una novedosa disciplina que pretende abordar problemas relacionados con el cáncer y explotar todas las herramientas matemáticas para mejorar el diagnóstico y optimizar los tratamientos.

En concreto, usaremos este campo para estudiar la Leucemia Linfoblástica Aguda (LLA), que es el cáncer que afecta a la formación de células sanguíneas encargadas de combatir las infecciones, y supone el 80 % de las leucemias en edad pediátrica. En los últimos años, la supervivencia de estos pacientes se ha visto incrementada considerablemente. No obstante, entre un 15 – 20 % de los tratamientos fracasan, lo cual lleva a una recaída en la enfermedad. Es por ello que se necesitan desarrollar nuevas estrategias que nos permitan conocer y seleccionar a los pacientes con mayor riesgo de recaída.

En este trabajo desarrollaremos modelos matemáticos evolutivos que describan el comportamiento de la enfermedad, estudiando la evolución de un clon leucémico usando datos de pacientes reales. Con esto se propone un modelo en el que poder ensayar futuras estrategias terapéuticas alternativas. La metodología usada combinará el estudio de ecuaciones diferenciales ordinarias, simulaciones numéricas, técnicas de análisis de datos y herramientas estadísticas con tratamiento de datos en los softwares *Python* y *Mathematica*.

Agradecimientos

Agradecer a mis padres los valores que me han inculcado y que han hecho de mí la persona que soy. A mis hermanos, Mariola, Antonio, Luismi y Adrián, gracias por formar la gran familia que somos. A mis sobrinos, que sin pretenderlo, me han ayudado en los momentos duros y han hecho que cada decisión que haya tomado la disfrutase como una niña.

A María, agradecerle las ganas de trabajar que me contagia cada día para luchar y avanzar hacia un mundo mejor. A Salvi, que siempre ha estado ahí, desde el comienzo, aportándome todo lo que precisase y desbloqueándome en cualquier momento que fuera necesario. Agradecer a Álvaro la ayuda que me ha proporcionado y que junto con María y Salvi, forman un equipo envidiable.

Por último, dar las gracias a todos los profesores y compañeros de clase que he tenido en el máster, porque de cada uno he podido aprender algo más allá de las matemáticas. En especial, a Rafa, que ha confiado en mí en todo momento y a Mamen que ha sido mi compañera de batalla en este camino de inolvidables historias.

Ana del Rosario Niño López

diciembre 2020

Índice general

1	Introducción	1
2	Conceptos previos de biología celular y cáncer	5
2.1	El cáncer	5
2.2	Hematopoyesis y linfopoyesis	7
2.3	Leucemia	8
3	Conceptos matemáticos previos	11
3.1	Sistemas de ecuaciones diferenciales	11
3.1.1	Problemas de valores iniciales y existencia de soluciones	12
3.2	Sistemas de ecuaciones diferenciales autónomos	13
3.2.1	Estabilidad de soluciones	14
3.2.2	Clasificación de sistemas autónomos	15
3.2.3	Solución analítica de sistemas lineales de orden n	15
3.3	Teorema de Hartman-Grobman y sus aplicaciones	16
4	Modelos matemáticos de hematopoyesis	19
4.1	Modelo de división asimétrica de células hematopoyéticas	19
4.2	Modelo de diferenciación de linfocitos tipo B en médula sana	25
5	Obtención y tratamiento de datos	31
5.1	Datos recogidos	31
5.2	Citometría de flujo	32
5.3	Análisis de los datos recogidos	34
5.4	Perfil inmunofenotípico de las leucemias linfoblásticas de la línea B	39
5.5	Resultados del tratamiento de datos	40

ÍNDICE GENERAL

5.5.1	Ubicación del clon.	40
5.5.2	Aproximación del diagnóstico.	42
6	Presentación y estudio del modelo	45
6.1	Modelo de desarrollo de un clon leucémico en médula	45
6.2	Estudio de las soluciones de equilibrio y estabilidad	47
6.3	Simulaciones para el modelo propuesto	51
6.4	Interpretación de los resultados	54
7	Resultados y Conclusiones	57
7.1	Aplicaciones de los resultados	57
7.2	Conclusiones	59
8	Ampliación y Proyectos Futuros	61
8.1	Planteamiento de proyecto futuro	61
A	Abreviaturas	63
B	Estabilidad y simulaciones de los modelos	65
B.1	Estudio de la estabilidad en los modelos de médula sana	65
B.2	Simulaciones de modelos de médula sana	68
B.3	Estudio de la estabilidad en los modelos de médula con clon leucémico	74
B.4	Simulaciones de modelos de médula con clon leucémico	82
C	Códigos	95
C.1	Algoritmo K-Means	95
C.2	Detección del clon	106
C.3	Clasificación de tipos de leucemia	113
	Bibliografía	117

Introducción

El cáncer es una de las enfermedades cuya incidencia está aumentando globalmente cada año. El número de casos de cáncer diagnosticados en España en 2020 alcanzará una cifra muy similar a la de 2019, con 277234 casos. En el mundo, supone unos 18.1 millones de casos nuevos [1]. Toda esta información proviene de los datos que son proporcionados por distintas organizaciones: Incidencia del cáncer en los cinco continentes, Incidencia Internacional de Cáncer Infantil, International Agency for Research on Cancer, la Organización Mundial de la Salud, etc. Estos datos, recogidos en [1], suponen una gran plataforma global para la información, el control y el estudio del cáncer.

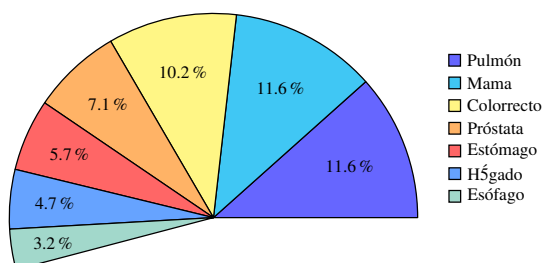


Figura 1.1: Tumores más frecuentemente diagnosticados en el mundo. Cifras del cáncer 2020 [2, 3].

En la Figura 1.1, podemos observar los tumores diagnosticados con más frecuencia en el mundo en ambos sexos. Se puede apreciar que los más comunes son el cáncer de pulmón y el cáncer de mama. Esta es una de las muchas gráficas que podemos encontrar, ya que la imagen cambiaría totalmente si tomamos los datos distinguiendo por sexo, país, tasas de mortalidad, etc.

1. INTRODUCCIÓN

En la representación de datos de la Figura 1.1 vemos que no aparece la leucemia como una de las enfermedades más comunes. En cambio, si analizamos los datos de cáncer en edad infantil, la representación cambia completamente. Según los datos de [4], se detectan cada año más de mil niños y niñas de entre 0 y 14 años con cáncer en España. El registro nacional de tumores infantiles, proporciona la información que se recoge en la Figura 1.2. Como vemos, la leucemia es el tipo de cáncer más frecuente en edad pediátrica.

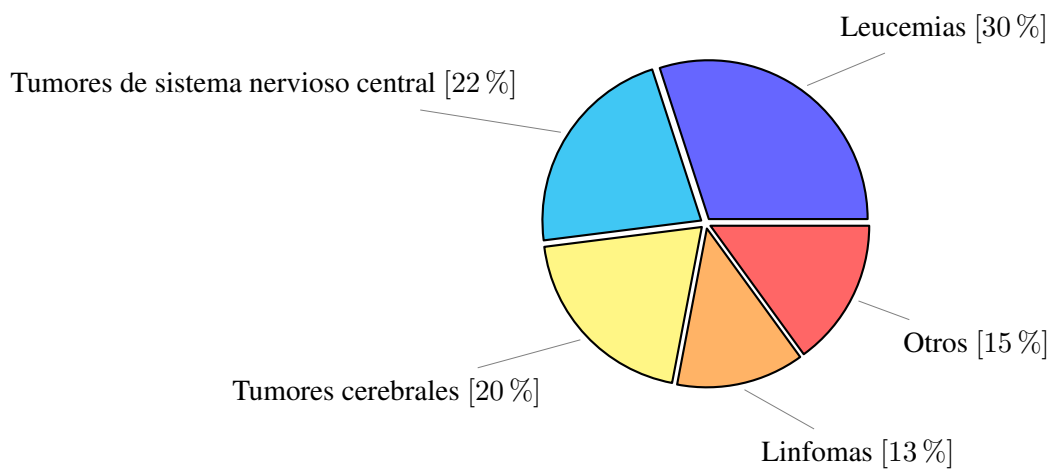


Figura 1.2: Tipos de cáncer más frecuentes en la población infantil. Información extraída de [4]. Imagen de autoría propia.

Algunos estudios muestran que la Leucemia Linfoblástica Aguda (LLA) constituye el 80 % de las leucemias en edad infantil [5, 6]. Como consecuencia de estos datos, cada vez son más los estudios referentes a los distintos procesos por los que se producen estas enfermedades. A pesar de la mejora de los resultados de los tratamientos en las últimas décadas, todavía un grupo de estos pacientes fracasan en el tratamiento [7, 8]. Es por esto que se necesitan desarrollar nuevos métodos para conocer y seleccionar a los pacientes con mayor riesgo de recaída. Para ello, se necesitará conocer el funcionamiento de los procesos donde se produce el fallo que desencadena la enfermedad.

Biológicamente, los órganos del cuerpo se forman por tejidos, y estos, a su vez, se componen de células. Si en algún proceso hay algún daño, las células se suicidan y otras toman sus componentes. Un cúmulo de daños hace que las células muten, es decir, cambien su ADN. Si estas células dañadas se reproducen, haciéndolo de manera anormal, en lugar de

morir, se produce lo que se denomina como cáncer.

Las células sanguíneas se forman por un proceso llamado hematopoyesis [9], que ocurre en la médula ósea. Las leucemias se originan en este proceso y para profundizar sobre ellas, es necesario conocer cómo se desarrollan tanto, cuando la médula está sana [10], como cuando se desarrolla el cáncer.

Los patrones que rigen el comportamiento de la biología de poblaciones se pueden presentar a través de modelos matemáticos. La biología matemática es un campo bastante reciente donde se combinan las bases de la biología y los modelos matemáticos que intentan reproducir los procesos para predecir su comportamiento [11]. Estos estudios están avanzando rápidamente en el campo de la oncología, dándose lugar una nueva disciplina: La Oncología Matemática [12, 13]. Se hace uso de las ecuaciones diferenciales, métodos numéricos, optimización o métodos estadísticos avanzados para la construcción de modelos matemáticos que describan los procesos en oncología.

En este trabajo se analizarán los modelos existentes en la literatura para médula sana, se tratarán datos de pacientes reales y se hará una propuesta de modelo en presencia de un clon leucémico. El estudio constará de las siguientes partes:

En el Capítulo 2 se realizará una introducción a los conceptos biológicos necesarios para la comprensión de nuestro estudio. Se presentarán los procesos de producción de células sanguíneas y se identificarán los fallos en los procesos con los distintos tipos de cáncer en sangre.

En el Capítulo 3 se mostrarán los principales resultados matemáticos utilizados en el estudio de ecuaciones diferenciales en las que se basan los modelos analizados.

En el Capítulo 4 se realizarán una selección de los modelos en la literatura más relevantes en nuestro trabajo.

En el Capítulo 5 se presentará la metodología llevada a cabo para el tratamiento de los datos, incluyendo el estudio de algoritmos de análisis de datos y presentando los resultados

1. INTRODUCCIÓN

obtenidos.

En el Capítulo 6 se propondrá un nuevo modelo que simula el comportamiento de la producción celular a partir de la aparición de un clon leucémico y se realizará el estudio y las simulaciones del mismo, integrando los datos del Capítulo anterior.

En el Capítulo 7 se incluirán las aplicaciones de los resultados obtenidos a lo largo de este trabajo y las conclusiones a las que se llega en el estudio.

En el Capítulo 8 se presentarán distintas propuestas de estudio que se podrán realizar en un proyecto futuro.

Se añaden además: El Anexo A con una guía de las abreviaturas empleadas; el Anexo B con el estudio de la estabilidad de los modelos presentados apoyados en el software *Mathematica* y sus simulaciones en *Python*; y el Anexo C, en el que se adjuntan los códigos de *Python* implementados para el desarrollo de algoritmos y nuevos métodos de tratamiento de datos.

Conceptos previos de biología celular y cáncer

En este capítulo se realizará una introducción al concepto de cáncer y su caracterización en el ámbito biológico. Se presentarán los procesos de formación de células sanguíneas y los tipos de enfermedad que desencadenan los errores producidos en estos procesos.

2.1 El cáncer

El cáncer es una enfermedad que se caracteriza por el desarrollo de células anormales que proliferan y se diseminan de forma incontrolada [2, 8, 14].

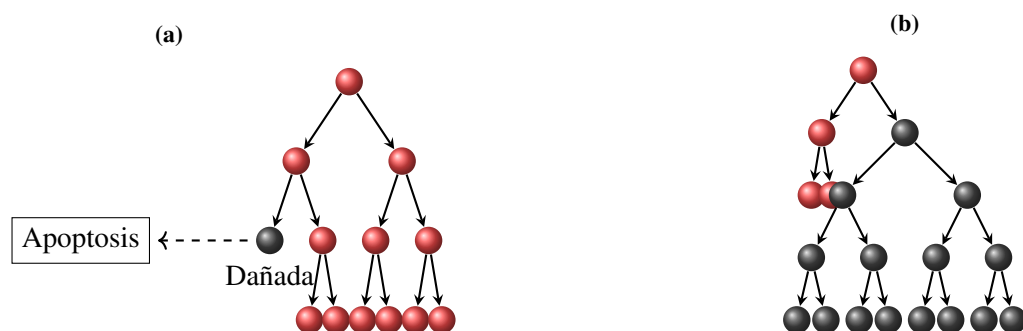


Figura 2.1: Comparación de comportamientos entre células sanas y células con cáncer. (a) Células sanas. (b) Célula cancerosa. Se representa de color negro la célula con daño irreparable. Imagen de autoría propia.

2. CONCEPTOS PREVIOS DE BIOLOGÍA CELULAR Y CÁNCER

Como podemos ver en Figura 2.1, mientras que las células normales se dividen y mueren de forma programada (apoptosis), las células cancerosas pierden la capacidad de morir si se producen fallos en el proceso. Es aquí donde se forman los llamados tumores o neoplasias, que pueden destruir los tejidos normales [2, 7].

El cáncer puede afectar a cualquier órgano del cuerpo, por lo que existen más de 100 tipos. Podemos clasificarlos según el tejido donde se formen [2, 7, 8]: Carcinomas, si afecta a las células epiteliales; sarcomas, si se trata de cánceres en el tejido conectivo; linfomas cuando se forman en tejido linfático; y leucemias, si se desarrollan en la médula ósea.

La carcinogénesis es el proceso de desarrollo del cáncer y se identifica con diez capacidades biológicas denominadas *hallmarks* o sellos distintivos del cáncer [15, 16]. Se ha sugerido que la mayoría de los tipos de cáncer adquieren un mismo conjunto de sellos distintivos durante su desarrollo, aunque la adquisición puede aparecer en diferentes fases del mismo [16, 17]. Podemos resumir los sellos distintivos como se muestran en la Figura 2.2, entre los que se encuentran la capacidad proliferativa ilimitada y la resistencia a la muerte celular, entre otras características. Estas características, son las que serán objeto de modelado matemático.



Figura 2.2: Sellos distintivos del cáncer. Imagen adaptada de [16].

2.2 Hematopoyesis y linfopoyesis

La hematopoyesis es el conjunto de mecanismos biológicos que intervienen en el proceso de producción de células sanguíneas que tiene lugar en la médula ósea [9, 18]. La médula ósea es un tejido suave y esponjoso que se encuentra en el centro de la mayoría de los huesos [7].

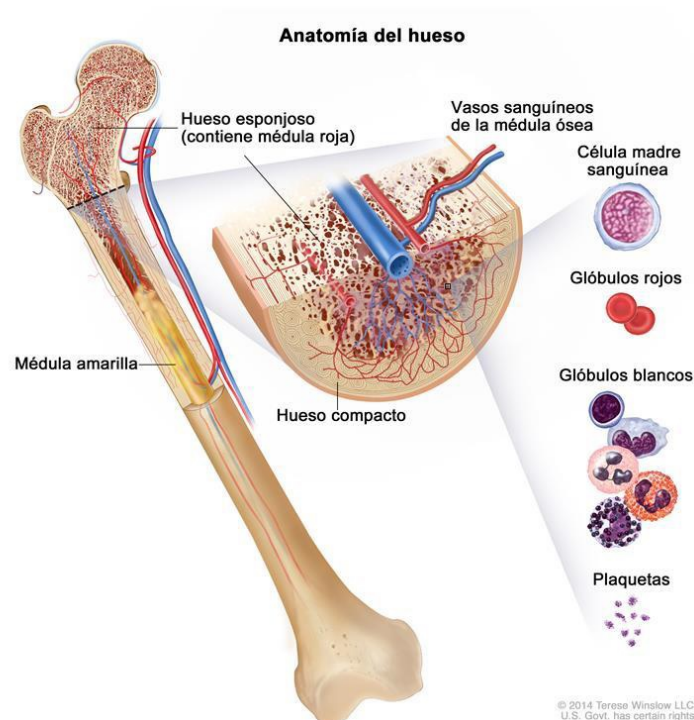


Figura 2.3: Anatomía de la médula ósea y del microentorno de producción celular. Imagen adaptada de [19].

Existen dos tipos de médula: La médula amarilla, que está compuesta en su mayoría, de grasa; y la médula roja, que contiene las células madre sanguíneas que se transforman en glóbulos rojos, glóbulos blancos o plaquetas. Cada una de estas células tiene una función asociada: Los glóbulos rojos se encargan de transportar el oxígeno desde los pulmones a todas las células del cuerpo; los glóbulos blancos, o leucocitos, ayudan a combatir infecciones; y las plaquetas sirven para formar coágulos y controlar los sangrados. Podemos ver en la Figura 2.3 que la médula ósea se encuentra en el centro de la mayoría de los huesos y tiene muchos vasos sanguíneos [8].

2. CONCEPTOS PREVIOS DE BIOLOGÍA CELULAR Y CÁNCER

La hematopoyesis es el proceso de diferenciación de células a partir de una célula madre [9, 18, 20], y se puede identificar con una estructura de árbol, como podemos ver en la Figura 2.4: La célula madre hematopoyética (HSC) puede tomar el linaje linfóide o mieloide. En cada linaje se producen distintas células sanguíneas a partir de su correspondiente célula progenitora. En la rama mieloide tiene lugar la producción de glóbulos rojos, plaquetas y mieloblastos y en la rama linfóide se da la formación de linfoblastos que a su vez producen linfocitos B, linfocitos T y linfocitos citotóxicos. Todos los cánceres en sangre se originan en disfuncionalidades que ocurren en este proceso [18].

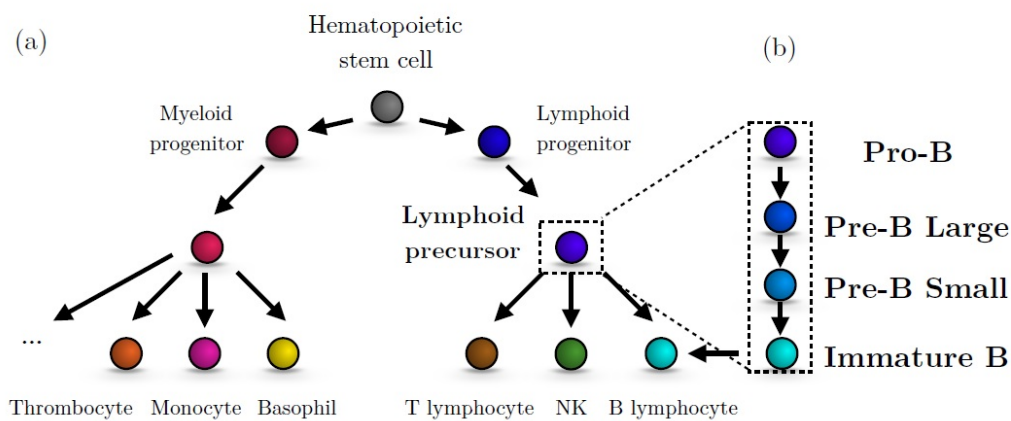


Figura 2.4: Esquema de diferenciación celular. (a) Representación esquemática de la hematopoyesis. (b) Proceso de maduración para células tipo B. Imagen tomada de [10].

Dentro del sistema hematopoyético, podemos definir la linfopoyesis como el proceso de desarrollo y maduración de las células de la rama linfóide. Este tipo de células tienen funciones principalmente asociadas con la secreción de anticuerpos, que se encargan de neutralizar agentes invasores [10, 21].

2.3 Leucemia

La leucemia es un tipo de cáncer que afecta a la sangre y médula ósea. Se origina en las células madre hematopoyéticas y se presenta por un aumento notable de los niveles de glóbulos blancos inmaduros que impiden la fabricación de glóbulos blancos normales. Esto dificulta la defensa ante posibles infecciones [7, 8, 22]. El conjunto de todas las células, de todos los estados madurativos, que llevan el mismo conjunto de alteraciones genéticas,

definen el clon leucémico [23]. Dependiendo de qué tipo de célula es la afectada, la leucemia será mieloide o linfoide. Además, se clasificará en aguda, si la célula es inmadura, o crónica, si se encuentra en el último paso de maduración [7]. El término aguda hace referencia a que la enfermedad avanza rápidamente y que se caracteriza por células leucémicas que no están totalmente diferenciadas, es decir, que son inmaduras. En cambio, la leucemia crónica avanza de forma más lenta y la proliferación se caracteriza por una cantidad mayor de células diferenciadas [8, 24]. De esta forma, podemos distinguir estos cuatro tipos:

- **Leucemia Linfoblástica Aguda (LLA):** Se produce en la rama linfoide y su desarrollo es rápido. Los pacientes más comunes son de edad pediátrica. Este tipo de enfermedad constituye el 80 % de las leucemias de edad infantil [6].
- **Leucemia Mieloide Aguda (LMA):** Afecta a las células de la rama mieloide y se desarrolla con rapidez. Es más común en personas de a partir de 65 años.
- **Leucemia Mieloide Crónica (LMC):** Influye en la producción de las células de la rama mieloide y se desarrolla con lentitud al principio. Más de la mitad de los pacientes son mayores de 65 años y afecta a un pequeño grupo de niños y adolescentes.
- **Leucemia Linfocítica Crónica (LLC):** Las células afectadas son de la rama linfoide y se desarrolla con lentitud. La mayoría de los pacientes tienen más de 65 años. Esta enfermedad afecta raramente a niños y adolescentes.

En nuestro caso, prestaremos atención a la Leucemia Linfoblástica Aguda (LLA), que es un cáncer formado por linfocitos inmaduros que impiden el crecimiento normal del resto de células de la sangre. Nos centraremos en la formación de linfocitos tipo B, denominando a este cáncer en sangre como LLA-B.

Dependiendo del tipo de LLA, su tratamiento, intensidad y duración, varían en cada paciente, y se regulan por protocolos muy elaborados [26, 27]. Es importante tener en cuenta los estados madurativos de los linfocitos tipo B, como se muestran en la Figura 2.5, presentada en [25]. Esta clasificación nos servirá como base en la realización y comprensión de modelos matemáticos. En el caso de LLA tipo B, se produce una proliferación exagerada de linfoblastos en algún estado madurativo de la célula: (I) Pro-B, (II) Pre-B, (III) Transición/Inmadura y (IV) Maduras. En la Figura 2.4 podemos ver de forma clara el proceso de diferenciación que se propone para linfocitos tipo B [10], hasta llegar a la sangre

2. CONCEPTOS PREVIOS DE BIOLOGÍA CELULAR Y CÁNCER

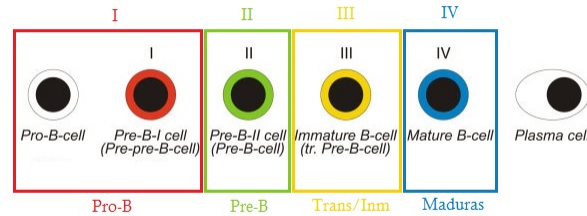


Figura 2.5: Estadios de diferenciación de linfocitos tipo B. Imagen adaptada de [25].

como células maduras y/o plasmáticas.

Antes de continuar el estudio, es conveniente aclarar algunos términos clave a los que se hacen referencia en los modelos que se van a presentar. Para distinguir las características de las poblaciones celulares que acabamos de indicar, se tienen en cuenta dos factores: La proliferación y la auto-renovación. La proliferación es el aumento de células como resultado del crecimiento y multiplicación celular [7]; y la auto-renovación es la capacidad de generación de nuevas células madre idénticas [28]. De esta forma, las características celulares de cada compartimento, se corresponderán con unos valores concretos para cada uno de estos términos, representados en el modelo en forma de parámetros.

Existen evidencias experimentales recientes que indican que el volumen de células leucémicas en pacientes individuales se componen de múltiples clones que llevan a diferentes mutaciones [29] y tienen distintas propiedades funcionales: tasas de proliferación y auto-renovación, entre otras [30]. Esta heterogeneidad implica un mayor obstáculo para la terapia y el manejo de la enfermedad. De hecho, muchos casos de recaídas de LLA y LMA se desencadenan por la selección de clones que han estado presentes de forma minoritaria en el momento del diagnóstico.

Todos estos comportamientos se modelizarán matemáticamente, considerando los datos y los procesos biológicos inherentes.

Conceptos matemáticos previos

La base del modelaje de procesos en biología son los sistemas de ecuaciones diferenciales, que se basan en potentes teorías matemáticas y pueden ser aplicadas para obtener grandes resultados. Los conceptos teóricos usados a lo largo de este trabajo se exponen a continuación tomando como base [31].

3.1 Sistemas de ecuaciones diferenciales

Sea $\mathbf{x} = x(t) = (x_1(t), x_2(t), \dots, x_n(t)) \in \mathbb{R}^n$ un vector con n componentes reales, $t \in \mathbb{R}$ la variable independiente, y $f = (f_1, \dots, f_n)$ una función vectorial tal que $f_i : U \subseteq \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$, para U abierto y $1 \leq i \leq n$. Consideramos la ecuación

$$\frac{d\mathbf{x}}{dt} = f(t, \mathbf{x}), \quad (3.1)$$

de la que, en otra notación se puede escribir como

$$\left(\frac{dx_1}{dt}, \dots, \frac{dx_n}{dt} \right) = (f_1(t, x_1, \dots, x_n), \dots, f_n(t, x_1, \dots, x_n)), \quad (3.2)$$

o equivalentemente, igualando por coordenadas,

3. CONCEPTOS MATEMÁTICOS PREVIOS

$$\begin{aligned}\frac{dx_1}{dt} &= f_1(t, x_1, \dots, x_n), \\ \frac{dx_2}{dt} &= f_2(t, x_1, \dots, x_n), \\ &\vdots \\ \frac{dx_n}{dt} &= f_n(t, x_1, \dots, x_n).\end{aligned}\tag{3.3}$$

Una solución del sistema (3.3) será $\Phi = (\Phi_1(t), \dots, \Phi_n(t))$, donde cada Φ_i , con $i = 1, \dots, n$, está definida en un intervalo abierto $I \subseteq \mathbb{R}$, con valores en \mathbb{R} , tal que satisface la ecuación $\frac{dx_i}{dt} = f_i(t, x_1, \dots, x_n)$.

Además, se tendrá que $\Phi = (\Phi_1, \dots, \Phi_n)$ es solución de (3.1), es decir, $\frac{d\Phi(t)}{dt} = f(t, \Phi(t))$.

3.1.1 Problemas de valores iniciales y existencia de soluciones

Dado el sistema (3.1), y una condición inicial en el instante t_0 tal que el valor de \mathbf{x} en t_0 sea $x_0 = (x_0^1, \dots, x_0^n) \in \mathbb{R}^n$, se define el problema de valores iniciales o problema de Cauchy asociado a (3.1) al sistema

$$\begin{cases} \frac{d\mathbf{x}}{dt} = f(t, \mathbf{x}), \\ \mathbf{x}(t_0) = x_0. \end{cases}\tag{3.4}$$

La solución del problema de valores iniciales (3.4) se denota como $\Phi(t; t_0, x_0)$ siendo Φ la solución de (3.1), definida en un intervalo maximal I , que cumple además que $\Phi(t_0) = x_0$.

Presentamos por tanto, un resultado que nos permite asegurar la existencia y unicidad de soluciones para problemas del tipo (3.4), su demostración podemos hallarla en [32].

Teorema 3.1. *Sea f una función continua en el punto $(t_0, x_0) \in U \subset \mathbb{R} \times \mathbb{R}^n$. Entonces, existe un entorno I_1 del punto t_0 suficientemente pequeño, tal que podemos asegurar la existencia en ese entorno de una solución $\phi_1 : I_1 \rightarrow \mathbb{R}^n$, de (3.4) satisfaciendo la condición inicial $\phi_1(t_0) = x_0$.*

Además, si $f \in \mathcal{C}^r$, con $r \geq 1$, y $\phi_2 : I_2 \rightarrow \mathbb{R}^n$ también solución (3.4) satisfaciendo la

3.2 Sistemas de ecuaciones diferenciales autónomos

misma condición inicial, entonces ambas funciones son iguales en el subconjunto $I_3 = I_1 \cap I_2$, es decir,

$$\phi_1(t) = \phi_2(t), \quad t \in I_3.$$

A lo largo del trabajo, en problemas tipo (3.4), tomaremos f como una función diferenciable con continuidad.

Definición 3.1. Llamaremos trayectoria de la solución a la gráfica de la función $\Phi(t; t_0, x_0)$, es decir, al conjunto de puntos $\{(t, x) \in \mathbb{R} \times \mathbb{R}^n : x = \Phi(t; t_0, x_0), t \in I\}$.

Definición 3.2. Las semiórbitas positivas y negativas asociadas al punto $x_0 \in \mathbb{R}^n$ se definen, respectivamente, como:

$$\gamma_{t_0}^+(x_0) = \{x \in \mathbb{R}^n : x = \Phi(t; t_0, x_0), t \in I, t \geq t_0\},$$

$$\gamma_{t_0}^-(x_0) = \{x \in \mathbb{R}^n : x = \Phi(t; t_0, x_0), t \in I, t \leq t_0\}.$$

Definición 3.3. La órbita de la solución será la unión de la semiórbita positiva y la semiórbita negativa:

$$\gamma_{t_0}(x_0) = \{x \in \mathbb{R}^n : x = \Phi(t; t_0, x_0), t \in I\}.$$

3.2 Sistemas de ecuaciones diferenciales autónomos

Los sistemas de ecuaciones que vamos a tratar en este trabajo serán sistemas autónomos, que son aquellos que se definen con $f(t, \mathbf{x}) = f(\mathbf{x})$, es decir, que no dependen de t de forma explícita. De esta forma, el sistema propuesto para el problema de valores iniciales (3.4) se expresará como sigue:

$$\begin{cases} \frac{d\mathbf{x}}{dt} = f(\mathbf{x}), \\ \mathbf{x}(t_0) = x_0. \end{cases} \quad (3.5)$$

Esta definición de f implica un estudio más sencillo del comportamiento de las soluciones del sistema. En el caso bidimensional, podremos hacer uso del plano de fases para representar las órbitas asociadas.

Definición 3.4. Sea \bar{x} un punto que verifica que $f(\bar{x}) = 0$ para el problema (3.5). Se tiene que \bar{x} es punto de equilibrio de la ecuación.

3. CONCEPTOS MATEMÁTICOS PREVIOS

Al ser $f(\bar{x}) = 0$, la variación de \mathbf{x} respecto a t , es decir, el cambio de \mathbf{x} es nulo. Por ello, se mantiene constante o en equilibrio.

Sea $\Phi(t)$ solución del problema de valores iniciales (3.5). Si $\Phi(t) = x_0$ para todo $t \in \mathbb{R}$ entonces x_0 es un punto de equilibrio por la definición 3.4 y la función $\Phi(t) = x_0$ será su correspondiente solución de equilibrio.

3.2.1 Estabilidad de soluciones

Veamos a continuación los tipos de estabilidad y el estudio gráfico de soluciones de un sistema. Esto nos servirá como herramienta para poder estudiar e interpretar los resultados.

Clasificación de soluciones

En un sistema diferencial, las soluciones se clasifican en estables, asintóticamente estables o inestables, dependiendo del comportamiento de las órbitas en el entorno que se esté analizando. Sea $\|\cdot\|$ la norma euclídea,

- diremos que $\Phi(t; t_0, x_0)$ es una solución estable si para todo $\varepsilon > 0$, existe $\delta > 0$ tal que, si $\|x'_0 - x_0\| < \delta$, entonces $\Phi(t; t_0, x'_0)$ existe, está definida para $t \geq t_0$ y verifica que $\|\Phi(t; t_0, x'_0) - \Phi(t; t_0, x_0)\| < \varepsilon$. Si no se verifica esta condición, se tratará de una solución inestable.
- una solución será asintóticamente estable si es estable y además, existe $\eta > 0$ tal que $\|\Phi(t; t_0, x'_0) - \Phi(t; t_0, x_0)\| \xrightarrow[t \rightarrow \infty]{} 0$ cuando $\|x'_0 - x_0\| < \eta$.

Se asumen los puntos de equilibrio como soluciones constantes, por lo que se tomarán las soluciones de equilibrio asociadas a los puntos para el estudio de su estabilidad.

Estudio gráfico de las soluciones y su estabilidad

Encontrar los puntos de equilibrio en el caso de los sistemas autónomos, consistirá en encontrar los puntos que anulen a la función f definida en el problema (3.5). En particular, cuando $f : U \subseteq \mathbb{R} \rightarrow \mathbb{R}$, podemos estudiar la estabilidad a partir de la gráfica de f . Esto es posible debido a que en un problema (3.5) para $n = 1$, la pendiente de la recta tangente a la curva $x(t)$ es su derivada en t , es decir $f(x(t))$. De esta forma, para los casos en los que el signo de f cambia de positivo a negativo, existirá un punto de equilibrio estable donde se anule f . En cualquier otro caso, se tratará de un punto de equilibrio inestable.

3.2.2 Clasificación de sistemas autónomos

Sea $A \in \mathcal{C}(I, \mathcal{M})$ y $b \in \mathcal{C}(I, \mathbb{R}^n)$, siendo $I \subset \mathbb{R}$ un intervalo abierto y $\mathcal{M} \subset \mathbb{R}^{n \times n}$. De forma general, se define un sistema lineal con la expresión

$$\frac{d\mathbf{x}}{dt} = A(t)\mathbf{x} + b(t), \quad (3.6)$$

donde la matriz $A(t)$ será la llamada matriz de coeficientes y el vector $b(t)$ el término independiente. Para $b(t) = 0$, el sistema (3.6) será un sistema lineal homogéneo y en caso contrario, no homogéneo.

Para el caso de sistemas autónomos, serán $A(t) = A$ y $b(t) = b$, siendo $A \in \mathcal{M}$, $b \in \mathbb{R}^n$, con componentes constantes definidas en \mathbb{R} . Así, el sistema (3.6) equivale a

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + b \quad (3.7)$$

En el caso de los sistemas no lineales autónomos, el sistema (3.6) vendría dado por la siguiente expresión:

$$\frac{d\mathbf{x}}{dt} = F(\mathbf{x}), \quad (3.8)$$

donde el término no lineal $F : D \rightarrow \mathbb{R}^n$ es una función de clase \mathcal{C}^1 en el abierto $D \subset \mathbb{R}^n$.

En estos casos, para el estudio del problema, se realizará el estudio de las soluciones obtenidas de la linealización del sistema. La linealización consiste en asociar un sistema lineal homogéneo de coeficientes constantes al sistema no lineal original, de manera que ambos sistemas se comporten de forma parecida en un entorno del punto a estudiar. Esta idea se desarrolla en el Teorema de Hartman-Grobman.

3.2.3 Solución analítica de sistemas lineales de orden n

Los sistemas de ecuaciones lineales de orden n de la forma (3.7), son los que principalmente se abarcarán a lo largo del estudio. Es por ello la importancia del Teorema 3.1 y su aplicación al problema que estudiamos.

Para visualizar el sistema de forma más sencilla, realizamos un cambio a notación matricial. Sea un sistema lineal de ecuaciones

$$X' = AX, \quad (3.9)$$

3. CONCEPTOS MATEMÁTICOS PREVIOS

siendo

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix},$$

con $a_{ij} \in \mathbb{R}$ para todo $i, j \in \{1, \dots, n\}$ y $X(t) = (x_1(t), \dots, x_n(t))^T$, de manera que

$$X'(t) = \left(\frac{dx_1}{dt}(t), \dots, \frac{dx_n}{dt}(t) \right)^T.$$

Las soluciones a un sistema de la forma (3.9) están definidas para cualquier $t \in \mathbb{R}$. Además, podemos hacer uso del *Principio de superposición*: Si $X_1(t), X_2(t), \dots, X_n(t)$ son soluciones de (3.9), entonces cualquier combinación lineal de estas soluciones también es una solución.

3.3 Teorema de Hartman-Grobman y sus aplicaciones

Consideremos la igualdad dada en el problema (3.5), $\frac{d\mathbf{x}}{dt} = f(\mathbf{x})$, siendo $\mathbf{x} \in \mathbb{R}^n$, Sea x_0 la órbita de una solución de equilibrio de (3.5). Se pretende determinar el comportamiento de las órbitas en un entorno de x_0 .

- Si se trata de un sistema lineal, se analiza un sistema equivalente cuya matriz de coeficientes sea la forma de Jordan asociada a la matriz de coeficientes del sistema.
- Si es no lineal, se aproxima $f(\mathbf{x})$ por una función lineal a través del desarrollo de Taylor de primer orden centrado en x_0 . Con lo cual, se modifica la ecuación que se representa en (3.8) quedando $F(x) \approx F(x_0) + DF(x_0)(x - x_0)$. Pero por construcción, se puede hacer que $F(x_0) = 0$, con lo cual, podemos reescribir la ecuación original de manera que, haciendo una translación en el eje, nos quede $\frac{d\mathbf{x}}{dt} = DF(x_0)\mathbf{x}$.

A continuación, presentamos el teorema de Hartman-Grobman que nos permitirá estudiar las soluciones de equilibrio obtenidas del sistema linealizado.

Teorema 3.2. (Hartman-Grobman). Si en las condiciones presentadas, $\text{Det}[Df(x_0)] \neq 0$ y $Df(x_0)$ no tiene autovalores complejos imaginarios puros, entonces existe un homeomorfismo $h : U \rightarrow \mathbb{R}^n$ definido en algún entorno U de x_0 que transforma las órbitas de la ecuación $\frac{d\mathbf{x}}{dt} = f(\mathbf{x})$ en las órbitas de (3.5). Además, puede ser elegido de tal forma que conserve el sentido de las órbitas y la parametrización del tiempo.

3.3 Teorema de Hartman-Grobman y sus aplicaciones

Este teorema garantiza la estabilidad o inestabilidad de soluciones a partir del estudio del comportamiento de un sistema linealizado y su demostración podemos encontrarla en [31]. De hecho, a partir de este teorema, se pueden obtener otros resultados que podemos usar para la aplicación directa en el estudio de estabilidad.

En resumen, y como herramienta que usaremos en los siguientes capítulos, supongamos f derivable con continuidad,

1. Si x_0 es punto de equilibrio de (3.5) y además, la matriz $Df(x_0)$ tiene todos sus autovalores con parte real estrictamente negativa, entonces la solución de equilibrio asociada a x_0 es una solución de equilibrio asintóticamente estable.
2. Si x_0 es punto de equilibrio de (3.5) y además, la matriz $Df(x_0)$ tiene todos sus autovalores con parte real estrictamente positiva, entonces la solución de equilibrio asociada a x_0 es una solución de equilibrio asintóticamente inestable.
3. Si alguno de los autovalores de la matriz $Df(x_0)$ tiene parte real positiva, entonces el punto de equilibrio x_0 del sistema es inestable.

De aquí se obtiene por tanto, que para asegurar la estabilidad asintótica de un punto de equilibrio en un problema (3.5), la matriz jacobiana del sistema tenga traza negativa y determinante positivo.

A partir de los conocimientos matemáticos presentados en este Capítulo, podremos realizar el estudio de los modelos matemáticos que representan procesos biológicos.

Modelos matemáticos de hematopoyesis

Entre los autores que han intentado modelar matemáticamente el crecimiento de distintos tipos de cáncer en sangre, podemos destacar el grupo *Applied Analysis and Modelling in Biosciences* dirigido por Anna Marciniak-Czochra. En los numerosos trabajos presentados, se plantean modelos matemáticos que estudian la población de células y el comportamiento de clones leucémicos en los procesos de hematopoyesis.

En este capítulo, se presentará uno de estos estudios [33] en el que se basa el modelo de diferenciación de linfocitos tipo B [10], también analizado en este capítulo. Todo esto servirá como base de modelos de linfopoyesis con la introducción de un clon leucémico, como veremos en capítulos próximos.

4.1 Modelo de división asimétrica de células hematopoyéticas

En esta sección, introduciremos las ideas básicas del modelo de división celular asimétrica presentadas en [33]. Se proponen tres modelos, suponiendo en todos ellos seis compartimentos como se muestra en la Figura 4.1. Los compartimentos representan el estado evolutivo de las células a lo largo del proceso de hematopoyesis, con lo que conllevan distintas propiedades de proliferación $\rho(t)$ y auto-renovación $a(t)$ para cada uno de ellos. Para simplificar el modelo, se tomarán ambas tasas constantes en el tiempo, siendo $\rho(t) = \rho$ y

4. MODELOS MATEMÁTICOS DE HEMATOPOYESIS

$a(t) = a$ en lo que sigue. La hipótesis en cada modelo varía en la existencia de una señal que afecta a la tasa de proliferación, a la tasa de auto-renovación o las dos simultáneamente.

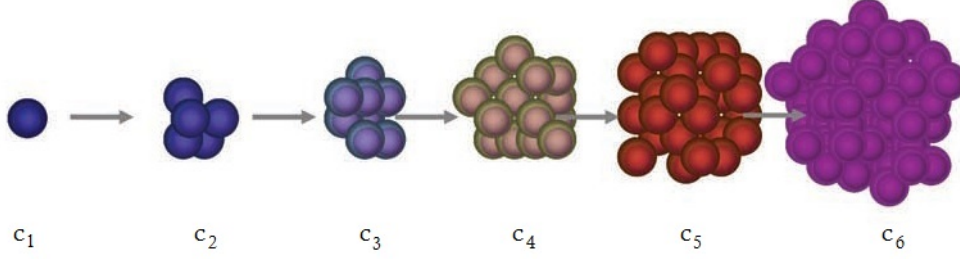


Figura 4.1: Concepto de proliferación de cada compartimento [33]. Nótese que en los primeros compartimentos, se considera una proliferación más lenta y va aumentando en cuanto se especifica el linaje que toma la célula.

En [33], se definen los compartimentos con la siguiente nomenclatura: Long-term repopulating stem cells, short-term repopulating stem cells, multipotent progenitor cells, committed progenitor cells, precursors, mature cells. Nos referiremos a ellos como c_1 , c_2 , c_3 , c_4 , c_5 , y c_6 , respectivamente, y tendremos en cuenta las propiedades que se presentan de cada uno. Las ecuaciones que modelan el comportamiento celular son ecuaciones diferenciales ordinarias. Las células se reproducen por mitosis, un proceso de división celular que consta de cuatro fases. Las células hematopoyéticas tienen una dinámica en su proliferación y auto-renovación específica para cada compartimento y que explicaremos a continuación.

Las células del primer compartimento (c_1) pueden entrar en división celular, con tasa ρ_1 , con lo cual, el flujo de mitosis es $\rho_1 c_1(t)$ y el flujo de células divididas por mitosis será $2\rho_1 c_1(t)$. Las células hija pueden renovarse y rellenar el compartimento o moverse al siguiente compartimento, por diferenciación, como se muestra en la Figura 4.2. La tasa de auto-renovación $a_1 \in [0, 1]$ implica que cada hija tenga la probabilidad de $1 - a_1$ de diferenciarse independientemente, i.e., pasar al siguiente compartimento. Así, la fracción de células hija cuya diferenciación es por división, será $1 - a_1$ y el flujo de células en la primera etapa es $2\rho_1 a_1 c_1(t)$. Tendremos que la tasa de cambio en $c_1(t)$ se da por la siguiente expresión:

4.1 Modelo de división asimétrica de células hematopoyéticas

$$\frac{dc_1}{dt} = 2\rho_1 a_1 c_1(t) - \rho_1 c_1(t) = [2a_1 - 1] \rho_1 c_1(t). \quad (4.1)$$

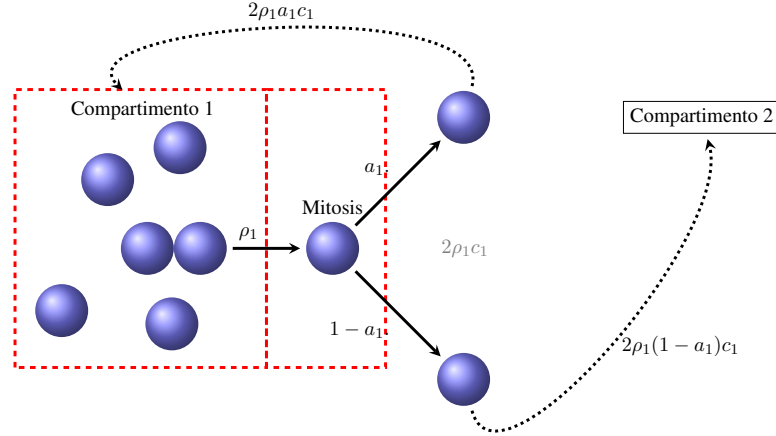


Figura 4.2: Descripción gráfica del flujo celular en el primer compartimento. Las células entran en mitosis con tasa ρ_1 y las células hijas pueden o bien auto-renovarse con probabilidad a_1 o bien pasar al siguiente compartimento con probabilidad $1 - a_1$.

Podemos agrupar los compartimentos de la etapa i , siendo $i = 2, \dots, 5$ con número de células $c_i(t)$ para describir su comportamiento. La diferenciación de células viene dada por un flujo de células de los compartimentos anteriores a los posteriores, es decir, de $i - 1$ a i . Para el paso de células de los siguientes compartimentos, tenemos en cuenta la forma en la que se comporta c_1 . El influjo de células por unidad de tiempo es $[2a_i - 1] \rho_i c_i$. A esto hay que añadirle las células que proceden del compartimento anterior con una tasa $1 - a_{i-1}$. Obtenemos que la evolución en el compartimento i es:

$$\frac{dc_i}{dt} = [2a_i - 1] \rho_i c_i(t) + 2[1 - a_{i-1}] \rho_{i-1} c_{i-1}(t), \quad i = 2, \dots, 5. \quad (4.2)$$

Por último, hay que tener en cuenta que en el compartimento de células maduras c_6 , las células no proliferan y tienen una tasa de muerte d , que consideraremos constante. Con lo cual, a este compartimento migran las células del compartimento anterior y se mueren dc_6 :

$$\frac{dc_6}{dt} = 2[1 - a_5] \rho_5 c_5(t) - dc_6(t). \quad (4.3)$$

4. MODELOS MATEMÁTICOS DE HEMATOPOYESIS

El comportamiento celular sin existencia de señal, podemos expresarlo por un sistema de ecuaciones como se presenta:

$$\begin{aligned}\frac{dc_1}{dt} &= (2a_1 - 1) \rho_1 c_1, \\ \frac{dc_2}{dt} &= (2a_2 - 1) \rho_2 c_2 + 2(1 - a_1) \rho_1 c_1, \\ \frac{dc_3}{dt} &= (2a_3 - 1) \rho_3 c_3 + 2(1 - a_2) \rho_2 c_2, \\ \frac{dc_4}{dt} &= (2a_4 - 1) \rho_4 c_4 + 2(1 - a_3) \rho_3 c_3, \\ \frac{dc_5}{dt} &= (2a_5 - 1) \rho_5 c_5 + 2(1 - a_4) \rho_4 c_4, \\ \frac{dc_6}{dt} &= 2(1 - a_5) \rho_5 c_5 - dc_6.\end{aligned}\tag{4.4}$$

Esta será la base de la que se partirá para cada modelo que se presenta a continuación. Las moléculas señalizadoras inducidas por el tratamiento, como la citoquina, que es una proteína que controla la producción celular [10], controlan la proliferación y la diferenciación de las células. Si asumimos α como la tasa máxima de segregación de citoquina, $\beta \geq 0$ la tasa de regulación provocada por las células maduras (c_6) y $\mu \geq 0$ la tasa que reduce los niveles de citoquina de manera proporcional a la cantidad existente, se obtiene como resultado:

$$\frac{dS}{dt} = \alpha - \mu S - \beta S c_6,\tag{4.5}$$

de donde, tomando $s = \frac{\mu}{\alpha} S$ y, para $\mu \neq 0$ y $k = \frac{\beta}{\mu}$, llegamos a

$$\frac{ds}{dt} = \mu(1 - s - k s c_6),\tag{4.6}$$

que expresa cómo la intensidad de la señal cambia a lo largo del tiempo. Como observación, el nivel de equilibrio para la señalización se consigue para el valor $\bar{s} = \frac{1}{(1 + k c_6)}$.

4.1 Modelo de división asimétrica de células hematopoyéticas

Como se puede apreciar, la señal máxima se alcanza en ausencia de células maduras y decrece asintóticamente a 0 a medida que crecen las células maduras.

Veamos cómo se proponen los distintos modelos según las hipótesis que se les asocia:

- Modelo I: La señal afecta a la proliferación mientras la tasa de auto-renovación no cambia en ninguno de los compartimentos.
- Modelo II: La señal influye en la tasa de auto-renovación, quedando fija la tasa de proliferación.
- Modelo III: Las tasas de auto-renovación y de proliferación se ven afectadas por la señal reguladora.

Presentamos a continuación el sistema resultante para el Modelo II, debido a que se adapta mejor tanto biológica como matemáticamente a la realidad del problema [33], siendo los otros modelos formulados análogamente y desarrollados en el mismo artículo.

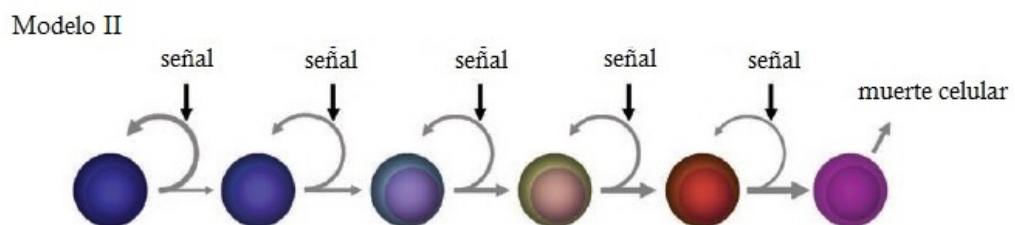


Figura 4.3: Influencia de la señal en la auto-renovación. Figura adaptada de [33].

4. MODELOS MATEMÁTICOS DE HEMATOPOYESIS

$$\begin{aligned}\frac{dc_1}{dt} &= [2a_1s(c_6) - 1] \rho_1 c_1, \\ \frac{dc_2}{dt} &= [2a_2s(c_6) - 1] \rho_2 c_2 + 2 [1 - a_1s(c_6)] \rho_1 c_1, \\ \frac{dc_3}{dt} &= [2a_3s(c_6) - 1] \rho_3 c_3 + 2 [1 - a_2s(c_6)] \rho_2 c_2, \\ \frac{dc_4}{dt} &= [a_4s(c_6) - 1] \rho_4 c_4 + 2 [1 - a_3s(c_6)] \rho_3 c_3, \\ \frac{dc_5}{dt} &= [2a_5s(c_6) - 1] \rho_5 c_5 + 2 [1 - a_4s(c_6)] \rho_4 c_4, \\ \frac{dc_6}{dt} &= 2 [1 - a_5s(c_6)] \rho_5 c_5 - dc_6, \\ s(c_6) &= \frac{1}{(1 + kc_6)}.\end{aligned}\tag{4.7}$$

Conclusiones sobre los modelos

En los distintos modelos propuestos en [33], en los que se tiene en cuenta una señal de re-alimentación influyente, se muestra una diferenciación de las células a diferentes compartimentos. Esto se corresponde con la realidad de médula sana, pero no ocurre para el caso de los clones leucémicos, ya que en el caso por ejemplo de la LLA, la capacidad de auto-renovación se pierde, al estar más diferenciados y alejados de las *Stem Cells*. Al contrario de lo que se expone como hipótesis en los modelos mostrados, la auto-renovación se da solo en el primer compartimento c_1 , debido a la cantidad de *Stem Cells* que se tienen en el compartimento considerado. Además, los contactos intracelulares favorecen el desarrollo de linfocitos y a medida que avanza el desarrollo de las células, se generan moléculas señalizadoras que regulan la transición de células entre etapas. En la médula ósea, solo un 15 % de las células originales consigue salir a la sangre.

4.2 Modelo de diferenciación de linfocitos tipo B en médula sana

Una vez estudiado el modelo presentado en la Sección 4.1, en el que se presentaba la dinámica de la diferenciación y comportamiento de las células madre, podemos basarnos en él para aplicar la misma filosofía al comportamiento de las células tipo B ya más diferenciadas, en el proceso que se presenta en la Figura 2.4. La estructura presentada anteriormente ha servido de apoyo para muchos otros estudios. En particular, en [23] se presenta una generalización de los modelos anteriores para n compartimentos y el desarrollo de un clon leucémico. Hay que tener en cuenta que la línea celular que es motivo de estudio en este apartado, está alejada de la célula madre hematopoyética, por lo que a medida que aumenta su estado madurativo, disminuye su potencial de autorenovación.

Consideramos ahora un sistema de tres compartimentos celulares con caracterización inmunofenotípica en términos de maduración: tempranas $C_1(t)$, intermedias $C_2(t)$ y tardías $C_3(t)$. Esta nomenclatura se correspondería en la caracterización de los estadios madurativos en la linfopoyesis como células Pro-B, Pre-B y de Transición, respectivamente. El modelo puede plantearse de la siguiente forma:

$$\begin{aligned}\frac{dC_1}{dt} &= \rho_1 C_1 - \alpha_1 C_1, \\ \frac{dC_2}{dt} &= \rho_2 C_2 + \alpha_1 C_1 - \alpha_2 C_2, \\ \frac{dC_3}{dt} &= \alpha_2 C_2 - \alpha_3 C_3.\end{aligned}\tag{4.8}$$

Si tenemos en cuenta el modelo (4.4) para tres compartimentos, podemos observar que esta formulación es equivalente a la presentada en las ecuaciones (4.8) siendo $\rho_i = \rho$, $\alpha_i = 2\rho_i(1 - a_i)$ para $i = 1, 2$ y $\alpha_3 = d$.

Este sistema necesita ser regulado por el mecanismo de señalización, que permite la interacción entre células. En este estudio se consideran dos hipótesis a estudiar: si la señal es producida por células tardías o por todas las células; y si la señal afecta a las tasas de proliferación o de transición. Podemos resumir los casos como se muestra en la Tabla 4.1.

4. MODELOS MATEMÁTICOS DE HEMATOPOYESIS

	Señal producida por células tardías	Señal producida por todas las células
Proliferación	Modelo A.1.	Modelo A.2.
Transición	Modelo B.1.	Modelo B.2.

Tabla 4.1: Tipos de modelos. Modelos diferenciados según la tasa a la que afecta la señal y el conjunto de células que influye en ella.

El estado de equilibrio de la señalización será el dado en la Sección 4.1 por la ecuación (4.5). Para el caso general en el que la señal viene dada por $T(t)$, obtenemos la siguiente expresión:

$$\bar{s}(t) = \frac{1}{1 + kT(t)}. \quad (4.9)$$

Ahora la expresión de $T(t)$ determina el tipo de células que participan en la inhibición. Consideramos por tanto dos casos:

1. $T(t)$ son solo las células tardías: $\bar{s}(t) = s_1(t) = \frac{1}{1 + kC_3}$.
2. Todas las células participan en la señalización: $\bar{s}(t) = s_2(t) = \frac{1}{1 + k \sum_{i=1}^3 C_i}$

Observemos que $\bar{s}(t)$ es una función decreciente y que podemos intuir que a mayor valor de k mayor capacidad de inhibición. Podemos finalmente obtener una familia de modelos de la siguiente forma:

$$\begin{aligned} \frac{dC_1}{dt} &= s_\rho \rho_1 C_1 - s_\alpha \alpha_1 C_1, \\ \frac{dC_2}{dt} &= s_\rho \rho_2 C_2 + s_\alpha \alpha_1 C_1 - s_\alpha \alpha_2 C_2, \\ \frac{dC_3}{dt} &= s_\alpha \alpha_2 C_2 - s_\alpha \alpha_3 C_3. \end{aligned} \quad (4.10)$$

Esto nos lleva a definir los cuatro modelos presentados anteriormente como sigue:

4.2 Modelo de diferenciación de linfocitos tipo B en médula sana

– Modelo A.1. ($s_\rho = s_1, s_\alpha = 1$)

$$\begin{aligned}\frac{dC_1}{dt} &= s_1\rho_1C_1 - \alpha_1C_1, \\ \frac{dC_2}{dt} &= s_1\rho_2C_2 + \alpha_1C_1 - \alpha_2C_2, \\ \frac{dC_3}{dt} &= \alpha_2C_2 - \alpha_3C_3.\end{aligned}\tag{4.11}$$

– Modelo A.2. ($s_\rho = s_2, s_\alpha = 1$)

$$\begin{aligned}\frac{dC_1}{dt} &= s_2\rho_1C_1 - \alpha_1C_1, \\ \frac{dC_2}{dt} &= s_2\rho_2C_2 + \alpha_1C_1 - \alpha_2C_2, \\ \frac{dC_3}{dt} &= \alpha_2C_2 - \alpha_3C_3.\end{aligned}\tag{4.13}$$

– Modelo B.1. ($s_\rho = 1, s_\alpha = s_1$)

$$\begin{aligned}\frac{dC_1}{dt} &= \rho_1C_1 - s_1\alpha_1C_1, \\ \frac{dC_2}{dt} &= \rho_2C_2 + s_1\alpha_1C_1 - s_1\alpha_2C_2, \\ \frac{dC_3}{dt} &= s_1\alpha_2C_2 - s_1\alpha_3C_3.\end{aligned}\tag{4.12}$$

– Modelo B.2. ($s_\rho = 1, s_\alpha = s_2$)

$$\begin{aligned}\frac{dC_1}{dt} &= \rho_1C_1 - s_2\alpha_1C_1, \\ \frac{dC_2}{dt} &= \rho_2C_2 + s_2\alpha_1C_1 - s_2\alpha_2C_2, \\ \frac{dC_3}{dt} &= s_2\alpha_2C_2 - s_2\alpha_3C_3.\end{aligned}\tag{4.14}$$

Simulaciones de los modelos

En esta sección se toma como referencia los datos con los que se presentan las simulaciones en [10]. Se reproduce en el software *Python* los modelos expuestos anteriormente y se analizan los resultados obtenidos. Además, nos apoyamos en la teoría presentada en el Capítulo 3 y con el software *Mathematica* realizamos el estudio de la estabilidad de las soluciones de equilibrio de los modelos tipo A, modelos (4.11) y (4.13), que son de los que se obtienen soluciones biológicamente viables, según [10]. El código implementado se muestra en el Anexo B.1. Las simulaciones realizadas para este modelo se adjuntan en el Anexo B.2 y se muestran los resultados en la Figura 4.4. Del estudio se obtiene lo siguiente:

4. MODELOS MATEMÁTICOS DE HEMATOPOYESIS

– Estabilidad en Modelo A.1.

– Estabilidad en Modelo A.2.

$$C_1 = 7.1565 \times 10^9,$$

$$C_2 = 8.70833 \times 10^{10},$$

$$C_3 = 2.6125 \times 10^{10}.$$

(4.15)

$$C_1 = 1.55331 \times 10^9,$$

$$C_2 = 1.89013 \times 10^{10},$$

$$C_3 = 5.67039 \times 10^9.$$

(4.16)

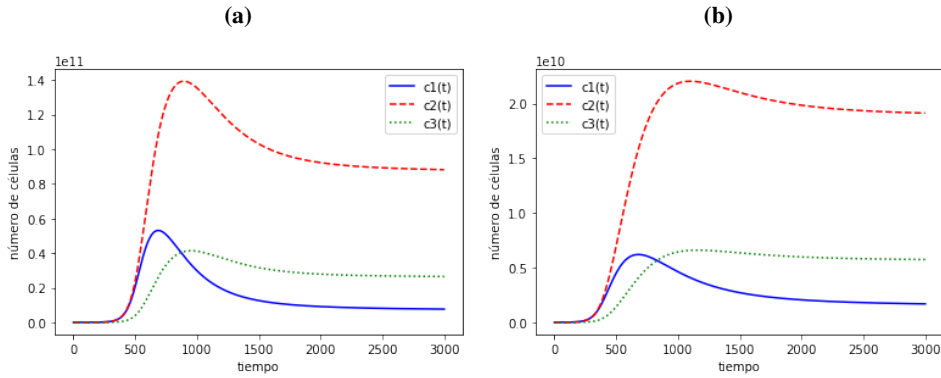


Figura 4.4: Evolución del número de células en función del tiempo. Se analizan y representan los compartimentos definidos en la familia de modelos 4.10 para los valores $\rho_1 = 0.0289h^{-1}$, $\rho_2 = 0.0193h^{-1}$, $\alpha_1 = 0.008h^{-1}$, $\alpha_2 = 0.006h^{-1}$, $\alpha_3 = 0.02h^{-1}$ y $k = 10^{-10}$. En ambos modelos, la gráfica de la solución de C_1 (en línea azul y continua), aunque comienza tomando valores más altos que el compartimento C_3 (línea de puntos verde), toma valores estables más bajos que los demás compartimentos. El compartimento C_2 (línea roja y discontinua) toma los valores más altos para las dos situaciones. **(a)** Modelo A.1. Observamos el crecimiento del número de células del compartimento C_1 y C_2 es más pronunciado y llega a un valor más alto **(b)** Modelo A.2. puede apreciarse el crecimiento más suave con valores más bajos. Imagen reproducida a partir de los datos encontrados en [10].

Conclusiones de los modelos

Según el análisis que se realiza en el estudio presentado, se observa que la señalización basada en la transición (Modelos B.1 y B.2) no es biológicamente viable, ya que en el estudio de la estabilidad, resultan inestables los puntos de equilibrio positivos no triviales. Así, se usan los modelos de señalización en la proliferación para las simulaciones numéricas, Modelos A.1 y A.2, mostrados en la Figura 4.4.

4.2 Modelo de diferenciación de linfocitos tipo B en médula sana

Como condiciones biológicas de este modelo y teniendo en cuenta la correlación estudiada para los parámetros, las células transitan más rápido desde las tempranas a las intermedias que de las intermedias a las tardías. De esta forma, podemos deducir que la población de células intermedias es más abundante, no por una mayor tasa de proliferación, sino por la lenta transición que le caracteriza. A partir de las simulaciones, también se tiene que la proporción de células es independiente de las células que se encargan de la señalización.

Las limitaciones del estudio viene dada por la elección de tres compartimentos para agrupar datos multidimensionales. La descripción se podría refinar para caracterizar de una forma más concreta los grupos de células. Además, los marcadores varían continuamente. Un modelo matemático con ecuaciones en derivadas parciales podría ser capaz de capturar esa variación.

Aunque simple, el modelo representa de forma realista el proceso de generación de células B representado en datos de médula ósea y nos lleva a un primer paso del estudio de fenómenos asociados al desarrollo de estas células.

Obtención y tratamiento de datos

En este capítulo, presentaremos los métodos llevados a cabo para la recogida de datos y obtención de información de una muestra de médula. El análisis de la muestra nos permite clasificar las células en los tres compartimentos presentados anteriormente según su estado madurativo. Así, se podrá tener un recuento de células aproximado para establecer modelos acordes a la realidad.

Los procesos llevados a cabo para el tratamiento de datos nos permitirá clasificar las células de una muestra, analizar su estado madurativo, hallar la proporción en la que se presentan las poblaciones de células en distintos pacientes y aproximarnos a algún modelo que rija el comportamiento de las células. Se incluirá también el estudio y caracterización de los tipos de leucemias.

5.1 Datos recogidos

Los datos en los que hemos basado el estudio han sido proporcionados por el Hospital Universitario Virgen del Rocío (Sevilla). El conjunto de datos se compone por un total de 38 pacientes, de los cuales, la información de las células que conforman la muestra de médula se presentan en ficheros tipo *.fcs*. Estas muestras se han tratado por citometría de flujo recogiendo información sobre la proporción en la que se presentan los distintos mar-

5. OBTENCIÓN Y TRATAMIENTO DE DATOS

cadore de las células.

Los datos para el estudio de leucemias linfoblásticas agudas están disponibles a través del estudio clínico retrospectivo LLAMATH, aprobado por el comité ético de Cádiz. Los datos clínicos de estos pacientes serán manejados bajo la actual Ley de Protección de Datos. Es por ello que se garantiza en todo momento el anonimato de los pacientes y la confidencialidad de los datos tratados en el estudio.

Los criterios de inclusión en el estudio fueron el diagnóstico de LLA entre febrero de 2009 y octubre de 2017, en edades comprendidas entre 1 y 19 años. Se excluyen de los datos aquellos pacientes de los que no se tienen un mínimo de 15 marcadores (IPT) en común con el resto de pacientes.

5.2 Citometría de flujo

La citometría de flujo es un método de análisis que permite la medición rápida de ciertas características de células suspendidas en líquido. Las proteínas asociadas a las células producen una señal de forma individual al interferir una fuente de luz [34, 35].

El citómetro de flujo hace pasar células en suspensión alineadas de una en una por delante de un haz luminoso. La información puede recogerse por la dispersión de la luz o por la emisión de luz por los fluorocromos unidos a ciertas proteínas de las células, al ser excitados por el rayo luminoso. Las señales luminosas detectadas se transforman en impulsos eléctricos que se amplifican y se convierten en señales digitales que son procesadas por un ordenador. Este proceso podemos verlo representado de manera esquemática en la Figura 5.1.

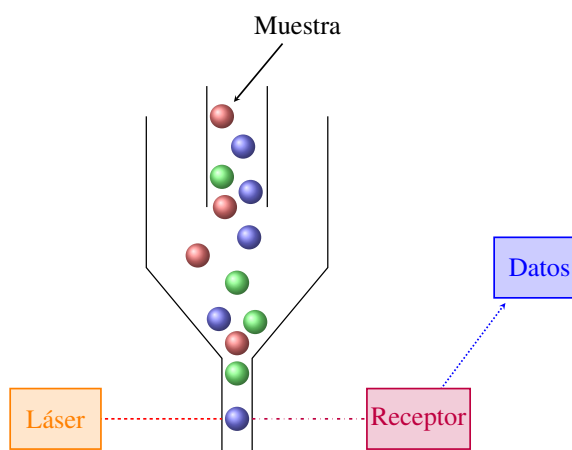


Figura 5.1: Esquema de citometría de flujo. La célula alcanzada por el láser emite una señal que el receptor recoge y codifica para la obtención de datos. Imagen de autoría propia.

Más concretamente, esta técnica mide la expresión de marcadores inmunofenotípicos en todas las células. Como cada estado de maduración se caracteriza por un conjunto de marcadores, podemos tener la proporción de células en cada etapa [10]. Los linfocitos tipo B es la población con marcador CD19 donde las células Pro-B, Pre-B y Transición presentadas en la Sección 4.2 corresponden con los marcadores $CD45^-/CD10^+$, $CD45^+/CD10^+$ y $CD45^+/CD10^-$, respectivamente, donde los superíndices indican la intensidad de fluorescencia. Podemos agrupar los datos en la Tabla 5.1 para su simplificación.

Tipo de Célula	CD45	CD10
Pro-B	-	+
Pre-B	+	+
Transición	+	-

Tabla 5.1: Marcadores característicos de los compartimentos celulares e intensidad de fluorescencia asociada.

Los datos obtenidos por citometría de flujo se almacenan en ficheros *.fcs* para su posterior tratamiento. El uso de *Python* nos facilita la lectura de los datos de los ficheros *.fcs* a través de las funciones implementadas para obtener un *Dataframe* (Anexo C.1). Los resultados obtenidos se muestran en la Figura 5.2 como ejemplo de obtención de algunas columnas del fichero.

	CD10	CD123	CD13	CD19	CD34	CD38	CD45
0	3.195855	2.957415	1.179513	2.751629	2.396975	2.542656	0.870945
1	2.372575	2.997735	1.219579	2.837649	2.458450	2.700136	1.444214
2	0.800262	2.411930	0.911228	3.092761	0.723521	1.803940	2.539877
3	1.127595	0.621219	0.947065	2.624190	1.018350	1.562567	2.641576
4	1.814947	3.114230	1.278458	2.957664	2.689724	2.822905	1.746378

Figura 5.2: Obtención de información de un fichero tipo *.fcs*. Se muestran los datos de 5 de las 51116 células correspondientes a la muestra tomada. El nombre de cada columna representa a cada marcador celular y los datos de cada fila se corresponden con cada célula de la muestra analizada.

A continuación, para obtener una nube de datos de la muestra que estamos tratando,

5. OBTENCIÓN Y TRATAMIENTO DE DATOS

podemos seleccionar los marcadores que deseemos y representar la composición de cada célula como se muestra en la Figura 5.3.

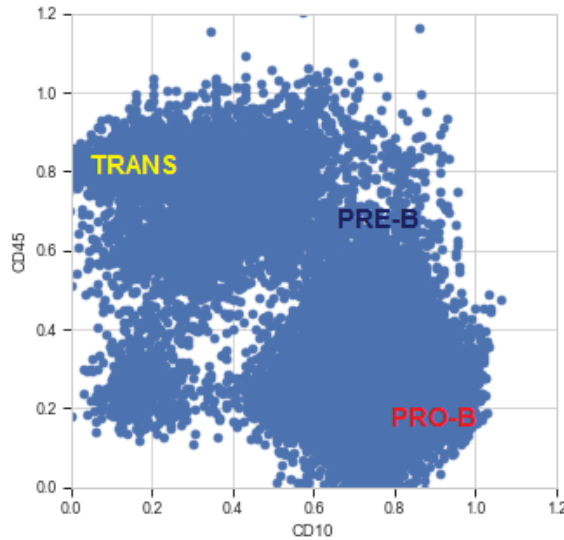


Figura 5.3: Nube de datos de una muestra de médula. Se representa la relación entre los marcadores CD10 y CD45 de una muestra que, analizada junto a la Tabla 5.1, se puede observar el reparto de las células según su estado madurativo: Las células más inmaduras corresponderán a valores mayores de CD10, mientras que un aumento de proporción de CD45 implicará compartimentos más maduros.

5.3 Análisis de los datos recogidos

Para analizar los datos obtenidos por citometría de flujo, hemos usado algoritmos de *Clustering* en el software *Python*. Clusterizar los datos consiste en la clasificación automática de estos, agrupándolos por categorías, y para ello usaremos la librería *Scikit-learn* de este software, [36]. Los *Clusters* no son más que subconjuntos disjuntos, o particiones del conjunto de datos a tratar.

Un mismo conjunto de datos puede ser clasificado de distintas formas, atendiendo a las propiedades, los grupos que queramos obtener y del algoritmo que se utilice para la clasificación. Los diferentes tipos de algoritmos disponibles (*Mean Shift*, *Spectral clustering*, *Gaussian mixtures* y *K-Means*, entre otros) pueden ser comparados entre ellos por la métrica usada, los parámetros necesarios para la función o su escalabilidad.

Algoritmo K-Means

Para el tratamiento de nuestros datos, usaremos el algoritmo *K-Means*, que destaca por su buena escalabilidad y ya habiendo sido usado para datos de citometría de flujo por otros autores [10, 37, 38].

Este algoritmo divide el conjunto de datos en k clusters, a los cuales se les asigna un centroide que no necesariamente es un dato de la muestra. El proceso busca los centroides que atiendan al criterio de mínimos cuadrados [39].

Sea $k \in \mathbb{N}$ el número de *clusters* elegido para la partición de un conjunto $D \subset \mathbb{R}^2$ de puntos $p_j = (p_j^x, p_j^y)$, para $1 \leq j \leq n$, con $j, n \in \mathbb{N}$. Sean c_i , $1 \leq i \leq k$, los centroides asociados a cada cluster C_i . El algoritmo consiste en minimizar la distancia euclídea entre los centroides y los puntos que pertenecen a su *cluster* a través de la función objetivo

$$\sum_{i=1}^k \sum_{j=1}^n \|p_j - c_i\|^2.$$

Definimos a continuación el proceso que lleva a cabo el algoritmo asociado a un ejemplo cuyo código se proporciona en Anexo C. Consideramos el ejemplo para $k = 3$, y un conjunto de puntos D . El proceso que toma el algoritmo es el siguiente:

- Etapa 0: Inicialización del algoritmo. Se designan los centroides c_i^0 , con $i = 1, \dots, k$ de forma aleatoria en \mathbb{R}^2 . Ver Figura 5.4a.
 - Asignación de *clusters*: A cada punto $p_j \in D$, se le asigna el cluster asociado al centroide más cercano, tal y como se observa en la Figura 5.4b. Esto es, se asigna el *cluster* C_m asociado al centroide c_m^0 tal que

$$\|p_j - c_m^0\| = \min_{1 \leq i \leq k} \|p_j - c_i^0\|$$

5. OBTENCIÓN Y TRATAMIENTO DE DATOS

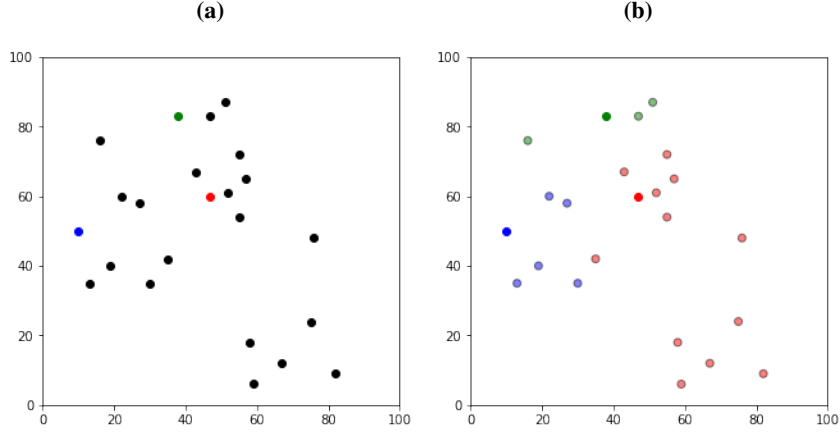


Figura 5.4: Etapa 0 del algoritmo de clasificación automática. (a) Etapa 0.1: Designación aleatoria de centroides c_1 , c_2 , c_3 coloreados en rojo, verde y azul, respectivamente. (b) Etapa 0.2: Asignación de puntos a cada cluster. A cada punto se le asocia el centroide que se encuentre a menor distancia.

- Etapa 1: Para cada *cluster* C_i , se ajusta un nuevo centroide c_i^1 que será el valor medio de los puntos incluidos en el cluster asociado, (ver Figura 5.5a). Tendremos para cada *cluster* C_i , $i = 1, \dots, k$, el centroide,

$$c_i^1 = \left(\sum_{p_j \in C_i} p_j \right) / \text{card}(C_i)$$

- Reasignación de *clusters*: Para cada punto se realiza de nuevo la asignación de *clusters* como en el segundo paso de la Etapa 0, Figura 5.5b.

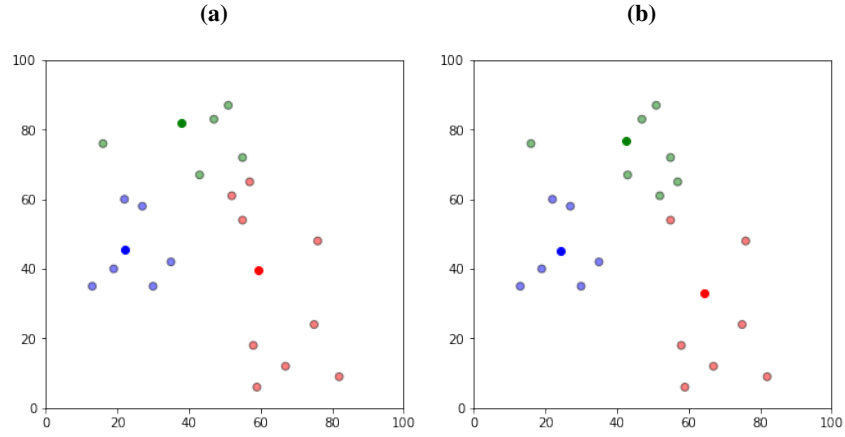


Figura 5.5: Etapa 1 del algoritmo de clasificación automática. (a) Reajuste de centroides de cada cluster. (b) Reasignación de clusters a cada punto.

- Repetición Etapa 1. Se repite el proceso hasta la Etapa q , siendo $q \in \mathbb{N}$ tal que para todo $i \in (0, k]$ se obtiene que $c_i^{q-1} = c_i^q$, es decir: El proceso termina cuando el reajuste de todos los centroides coincide con los centroides de la etapa anterior.

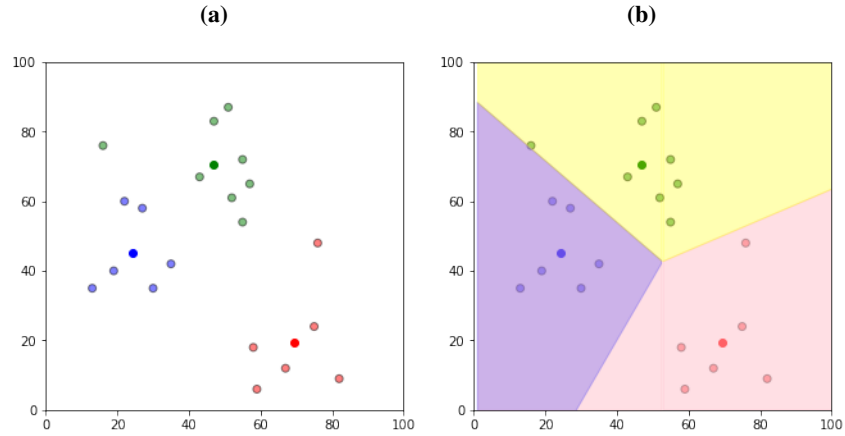


Figura 5.6: Asignación final de clusters. (a) Centroides en la última etapa. A partir de la repetición del algoritmo, llegamos a una etapa donde los centroides recalculados son los mismos que en la etapa anterior. (b) Clusters asociados a cada centroide. Se muestra la partición del plano dada por las mediatrices de los segmentos formados por cada par de centroides.

Una vez comprendido el proceso que se lleva a cabo la función *K-Means* para *clusterizar* los datos, podremos realizar una clasificación automática de las muestras usando

5. OBTENCIÓN Y TRATAMIENTO DE DATOS

el paquete *Scikit-learn* de *Python*. Este proceso de clasificación nos será útil para determinados casos en los que podemos discutir su aplicabilidad gracias a la comprensión del proceso que se lleva a cabo en el algoritmo. Se adjunta ejemplo de tratamiento de datos con esta función en Anexo C.1.

Ejemplo de clusterización de una población celular

Tomando ahora la muestra representada en la Figura 5.3, aplicamos el algoritmo *k-means* para obtener una clasificación de las células según sus marcadores CD10 y CD45. Podemos observar el resultado en la Figura 5.7. Además, podemos también contar el número de células que pertenecen a cada compartimento. Esta función se muestra en el Anexo C.1 y nos será útil para determinar las proporciones en las que se presenta cada estado madurativo en la muestra.

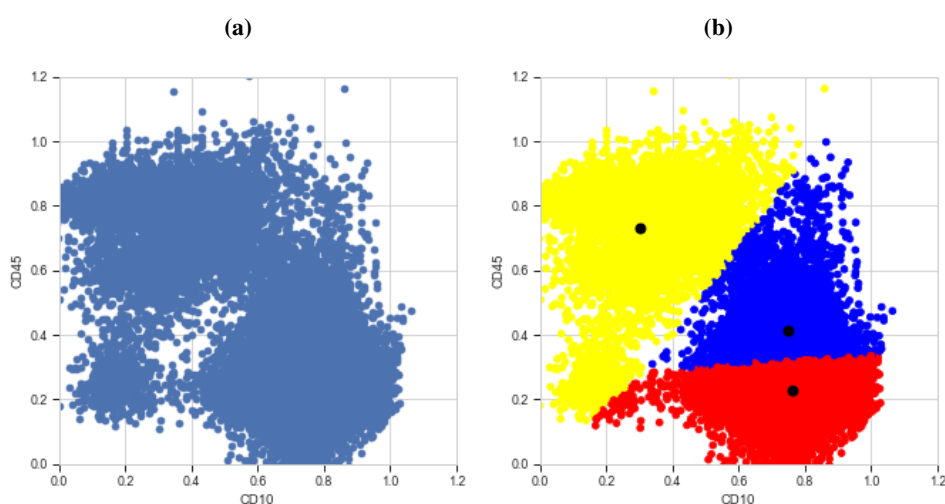


Figura 5.7: Comparativa de una nube de datos y su clusterización. Clasificación de las células de una muestra de médula atendiendo a sus marcadores CD10 y CD45 para obtener los compartimentos según su estado madurativo. (a) Nube de datos mostrada en Figura 5.3. (b) Nube de datos clusterizada. Se ha usado el algoritmo *k-means* para su clasificación. La población de color rojo corresponde a las células del compartimento Pro-B, ($CD45^-/CD10^+$), las células tipo Pre-B son las coloreadas en azul y son células ($CD45^+/CD10^+$) y por último, el compartimento de Transición se asocia a células ($CD45^+/CD10^-$), coloreadas en amarillo.

5.4 Perfil inmunofenotípico de las leucemias linfoblásticas de la línea B

Después de una extensa revisión bibliográfica, se debe aclarar que los nombres de los tipos de leucemia, no corresponden a fallos en el compartimento que lleva su mismo nombre. Esto puede llevar a confusión en el seguimiento del trabajo. Hasta ahora hemos realizado una clasificación de las células en los compartimentos asociados a la linfopoyesis y a continuación vamos a analizar el perfil inmunofenotípico de los tipos de leucemias. Es decir, estudiaremos los principales marcadores que diferencian las leucemias en la línea B que pueden ser diagnosticadas. Para estas caracterizaciones tomaremos como referencia [40].

Podemos realizar por lo tanto, una clasificación de los pacientes atendiendo a los marcadores asociados a cada tipo de leucemia:

- Las leucemias tipo Pro-B se corresponden con células (CD10⁻/CD34⁺/CD38⁺⁺/CD20⁻).
- Las leucemias tipo Común se asocian a células (CD10⁺⁺/CD34[±]/CD38⁺/CD20[±]).
- Las leucemias tipo Pre-B se caracterizan por células (CD10⁺/CD34⁻/CD38[±]/CD20⁺).
- En las leucemias tipo Maduras las células son (CD10[±]/CD34⁻/CD38[±]/CD20⁺).

Toda esta información se recoge en la Tabla 5.2. Entre nuestros datos, no tenemos pacientes diagnosticados con leucemias linfoblásticas tipo Maduras, con lo cual, no nos centraremos en diferenciar este tipo del tipo Pre-B. Por lo tanto, no atenderemos al marcador C_{μ} que se muestra en la Tabla 5.2. Los marcadores elegidos son los que muestran diferencias entre los tipos de leucemia según los datos que encontramos en [40]. Por ejemplo, el marcador CD79a se presenta de la misma forma en todas las células en cada tipo de leucemia, con lo cual, es irrelevante para distinguir entre ellos y no se incluye en esta tabla.

Tipo Leucemia	CD10	CD34	CD38	CD20	C_{μ}
Pro-B	-	+	++	-	-
Común	++	±	+	±	-
Pre-B	+	-	±	+	+
Maduras	±	-	±	+	-

Tabla 5.2: Marcadores celulares para cada tipo de Leucemia. Resumen de marcadores característicos de las células para cada tipo de leucemia y su intensidad de fluorescencia. Tabla modificada tomada de [40].

5. OBTENCIÓN Y TRATAMIENTO DE DATOS

Podemos comparar las nubes de datos obtenidas de las muestras atendiendo a los distintos marcadores indicados anteriormente, como se muestra en la Figura 5.8.

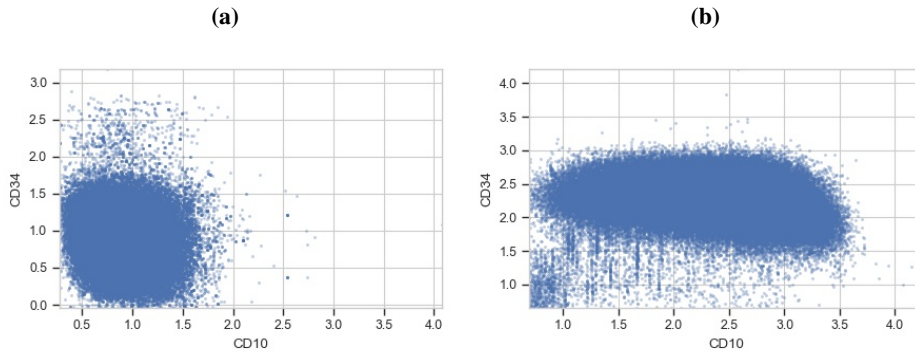


Figura 5.8: Comparativa de nube de datos entre Leucemia Pro-B y Leucemia Común.

Se representan los datos de la muestra de dos pacientes atendiendo a los marcadores CD10 y CD34 que diferencian los tipos de leucemias diagnosticadas. (a) Muestra de paciente con diagnóstico Pro-B. Observamos cómo las células se agrupan y se aglomeran en valores muy bajos de CD10. (b) Muestra de médula de paciente con diagnóstico Común. Los valores de CD10 van aumentando y a su vez se asocian con valores mayores de CD34.

5.5 Resultados del tratamiento de datos

Gracias a la programación en *Python* de distintas funciones de lectura de datos, se llega a una mejora en la comprensión de la información recogida por el citómetro de flujo lo que conduce a los resultados explicados a continuación.

5.5.1 Ubicación del clon.

Los datos de los ficheros *.fcs* nos indican qué proporción de células poseen cierta intensidad de los marcadores CD10 y CD45. Como ya se presentaban antes los distintos estados madurativos, las células con mayor cantidad de CD10, serían las más inmaduras, es decir, las células tipo Pro-B. De esta forma, podemos realizar un estudio a cada muestra clasificando las células en Pro-B, Pre-B y Transición, según la proporción en la que se encuentren los marcadores indicados. Así, para cada muestra se hará uso del algoritmo de clasificación automática *k-means* y podremos definir las tres poblaciones celulares que estamos estudiando según sus marcadores.

El clon leucémico necesita ser ubicado en un compartimento celular para asociarle las propiedades que le correspondan en relación a la proliferación y la auto-renovación. Para hallar en qué estadio madurativo se encuentra el clon, se ha propuesto un estudio de las anomalías que se dan entre las proporciones de población Pro-B, Pre-B y Transición, en relación a los datos y los modelos estudiados de médula sana.

Detección del estado madurativo del clon

A partir de los estudios de células en los compartimentos Pro-B, Pre-B y Transición en los modelos de médula sana [10], se tiene que en un tiempo $t > 125$ días se alcanza un estado estable en las proporciones de células de cada compartimento. Del análisis de los modelos ya realizado, podemos deducir que para médula sana, las proporciones en las que se halla cada compartimento cumplen lo siguiente: El compartimento de células Pre-B se encuentra en una mayor proporción que el compartimento de células de Transición, y el compartimento con menos células es el Pro-B. Esto es, que las proporciones en médula sana cumplen $|\bar{c}_2| > |\bar{c}_3| > |\bar{c}_1|$, siendo \bar{c}_i el estado de equilibrio de cada compartimento. Por lo tanto, para las muestras que no cumplan este orden, se tendrá un clon leucémico en el estado madurativo que presente una proporción anormal de células.

Con los datos obtenidos en los resultados (4.15) y (4.16), podemos calcular las proporciones en las que encontramos cada compartimento cuando se alcanza estabilidad en la médula sana. Las proporciones serán las mismas, independientemente del modelo por el que nos rijamos, y de manera aproximada son: 6 % para las Pro-B, 72 % para las Pre-B y 22 % para las de Transición.

Este proceso de comparación se ha llevado a cabo en *Python* para las muestras que tenemos disponibles como se muestra en el Anexo C.2. El código nos permite identificar qué compartimento tiene una proporción anormal de células, y por tanto, el compartimento del que el clon toma las características celulares. En la Figura 5.9 se muestra el resultado obtenido de calcular la proporción de células que se tiene en cada compartimento para cada paciente. El código completo se adjunta en el Anexo C.2.

5. OBTENCIÓN Y TRATAMIENTO DE DATOS

Pacientes	Total_de_células	Pro-B	Pre-B	Transición
0	156962	0.499006	0.473650	0.027344
1	162399	0.466025	0.507245	0.026730
2	161263	0.287140	0.360176	0.352685
3	109575	0.427004	0.441405	0.131590
4	165335	0.204004	0.327686	0.468310
5	162135	0.470855	0.507737	0.021408
6	154432	0.304380	0.215363	0.480257
7	93462	0.557521	0.386339	0.056140
8	150673	0.419027	0.398014	0.182959
9	143987	0.371165	0.550890	0.077945
10	139709	0.310202	0.640617	0.049181
11	161599	0.510900	0.459718	0.029381
12	70144	0.674940	0.070099	0.254961
13	124501	0.381788	0.562831	0.055381
14	164261	0.312990	0.319157	0.367854

Figura 5.9: Proporciones de células en cada compartimento. Se muestra como ejemplo el estudio de las muestras de 15 pacientes, a las que se les ha aplicado el algoritmo *k-means* y posteriormente, se han calculado la proporción de células que pertenece a cada compartimento.

5.5.2 Aproximación del diagnóstico.

La lectura de los ficheros también nos permite centrarnos en otros marcadores además de los CD10 y CD45. Como se recoge en [40], existen distintas caracterizaciones para identificar el tipo de leucemia que tiene el paciente. Los tipos de leucemia serán Pro-B, Comunes, Pre-B y Maduras. Para cada paciente se ha analizado la muestra y las proporciones de los marcadores que son significativos para cada tipo de leucemia, presentados en la Sección 5.4. De esta forma, podemos proponer una clasificación según los datos obtenidos que se aproxima bastante a los diagnósticos de los pacientes. Los resultados obtenidos son presentados en el Anexo C.3.

De la clasificación propuesta de los pacientes según su tipo de leucemia, hemos obtenido un resultado equivalente al diagnóstico asociado al paciente en 29 de los 38 casos. Esto nos lleva a un acierto del 76.31 %, proceso que habrá que matizar cuando aumente el número de muestras disponibles y la información recibida de las mismas. El resumen de los resultados se recogen en la Tabla 5.3. En ella, las filas corresponden al diagnóstico que hemos estimado para los pacientes tras el tratamiento de los datos y en las columnas, el diagnóstico real. Podemos observar que para el grupo de diagnóstico Pro-B se aproximan erróneamente 2 pacientes; para el grupo con diagnóstico de tipo Común, 2 pacientes; y para los pacientes diagnosticados con tipo Pre-B, hemos clasificado de forma fallida a 5

5.5 Resultados del tratamiento de datos

pacientes. Esto supone un acierto del 91.66 % en el diagnóstico de leucemia tipo Común, del 54.54 % del tipo Pre-B y del 33.33 % de las leucemias tipo Pro-B.

Diag. Aprox. \ Diag. Real	Pro-B	Común	Pre-B
Pro-B	1	0	0
Común	0	22	5
Pre-B	2	2	6
Total de diagnósticos	3	24	11

Tabla 5.3: Comparativa de la clasificación de pacientes y su diagnóstico. Resultados obtenidos a través del tratamiento de los datos. Se destacan en negrita los pacientes en los que coinciden nuestra clasificación y su diagnóstico.

Presentación y estudio del modelo

Como ya venimos estudiando a lo largo de este trabajo, las leucemias son enfermedades malignas del sistema hematopoyético [23]. De forma similar que en el sistema hematopoyético sano, un clon leucémico se organiza en múltiples estados madurativos jerarquizados [41]. Existen evidencias teóricas y experimentales que sugieren que la tasa de proliferación y auto-renovación tienen un impacto significativo en la dinámica de la enfermedad y el pronóstico del paciente. Es por ello, que se presenta un modelo en el que el clon se relaciona con uno de los compartimentos ya presentados, donde adquiere unas características propias asociadas al estado madurativo de su ubicación.

En este capítulo se propone un modelo de la dinámica de los linfocitos tipo B de linfocitos tipo B y el clon leucémico, se realiza un estudio de su estabilidad y se simulan los distintos casos. Esto nos conduce finalmente a la interpretación de los resultados. Se consideran nuevamente los estados madurativos en el proceso de linfopoyesis: Pro-B, Pre-B y Transición.

6.1 Modelo de desarrollo de un clon leucémico en médula

Para nuestra propuesta, tendremos en cuenta la familia de modelos (4.10) para médula sana presentada en la Sección 4.2, en su particularización a los modelos A, donde la señalización afecta a la proliferación. Es decir, tomamos la familia de modelos (4.10) con $s_\alpha = 1$

6. PRESENTACIÓN Y ESTUDIO DEL MODELO

y añadimos el compartimento $L(t)$ en el que se encuentran las células leucémicas. Incluiremos el clon leucémico asumiendo la no diferenciación a otros compartimentos y su influencia en la proliferación de las células sanas a través de la señalización.

Tomamos un crecimiento logístico para el compartimento $L(t)$. En el caso de la capacidad de carga del compartimento de células leucémicas, no tomaremos el soporte k que venimos suponiendo para las demás células de la médula porque sería biológicamente inviable. Por tanto, tomaremos una capacidad máxima de células leucémicas $L_{\text{máx}}$. Por otro lado, consideraremos γ_L una tasa de muerte celular para las leucémicas.

Es importante tener en cuenta que las células leucémicas tomarán una tasa de proliferación ρ_L asociada al estadio madurativo en el que se encuentre el clon, esto es,

$$\rho_L = \rho_1, \text{ si el clon apareciera en las células Pro-B,}$$

$$\rho_L = \rho_2, \text{ si se diese en células del estadio Pre-B.}$$

De forma general, podemos proponer un flujo en el compartimento celular del clon de la siguiente manera, teniendo en cuenta una señalización s_L que afecta a la proliferación del mismo:

$$\frac{dL}{dt} = s_L \rho_L L \left(1 - \frac{L}{L_{\text{máx}}} \right) - \gamma_L L.$$

Las células leucémicas se verán afectadas en su proliferación por las células sanas pero no por las propias células leucémicas, es decir, que existe una señalización particular para el clon que será

- En el caso en el que solo se tienen en cuenta las células del último compartimento

$$s_{L1}(t) = \frac{1}{1 + kC_3},$$

- y si todas las células participan en la señalización

$$s_{L2}(t) = \frac{1}{1 + k \left(\sum_{i=1}^3 C_i \right)}$$

6.2 Estudio de las soluciones de equilibrio y estabilidad

La señalización es análoga a la presentada en los modelos de médula sana (4.10). En el caso de la proliferación de células sanas habrá que tener en cuenta la influencia de las leucémicas en la señal, obteniéndose así, dos tipos de señalización:

- Solo se tienen en cuenta las células de transición y las leucémicas $s_1(t) = \frac{1}{1 + k(L + C_3)}$
- Todas las células participan en la señalización $s_2(t) = \frac{1}{1 + k\left(L + \sum_{i=1}^3 C_i\right)}$.

De esta manera, el modelo que proponemos para el estudio es el que sigue:

$$\begin{aligned}\frac{dC_1}{dt} &= s_\rho \rho_1 C_1 - \alpha_1 C_1, \\ \frac{dC_2}{dt} &= s_\rho \rho_2 C_2 + \alpha_1 C_1 - \alpha_2 C_2, \\ \frac{dC_3}{dt} &= \alpha_2 C_2 - \alpha_3 C_3. \\ \frac{dL}{dt} &= s_L \rho_L L \left(1 - \frac{L}{L_{\text{máx}}}\right) - \gamma_L L\end{aligned}\tag{6.1}$$

Denotaremos Modelo I al que considera $s_L = s_{L1}$ y $s_\rho = s_1$, mientras que Modelo II tendrá $s_L = s_{L2}$ y $s_\rho = s_2$. En cada uno de los modelos habrá dos casos a tener en cuenta dependiendo de dónde se ubique el clon leucémico, lo cual determinará el parámetro ρ_L .

6.2 Estudio de las soluciones de equilibrio y estabilidad

A partir de la teoría presentada en el Capítulo 3, podremos analizar el modelo propuesto e interpretar los resultados obtenidos. Para ello, realizamos un estudio de los puntos de equilibrio del modelo y su estabilidad apoyados en el software *Mathematica*. En el Anexo B.3, se muestra el código ejecutado para un estudio teórico de la estabilidad de los puntos de equilibrio. En primer lugar, buscamos los puntos de equilibrio del sistema. Para hallarlos, igualamos a 0 el flujo de cada compartimento:

6. PRESENTACIÓN Y ESTUDIO DEL MODELO

$$\begin{aligned}
 0 &= s_\rho \rho_1 C_1 - \alpha_1 C_1, \\
 0 &= s_\rho \rho_2 C_2 + \alpha_1 C_1 - \alpha_2 C_2, \\
 0 &= \alpha_2 C_2 - \alpha_3 C_3. \\
 0 &= s_L \rho_L L \left(1 - \frac{L}{L_{\text{máx}}}\right) - \gamma_L L
 \end{aligned} \tag{6.2}$$

Soluciones teóricas del Modelo I ($s_\rho = s_1$ y $s_\rho = s_{L1}$):

- $(0, 0, 0, 0)$
- $\left(0, 0, 0, \frac{L_{\text{máx}}(\rho_L - \gamma)}{\rho_L}\right)$
- $\left(\frac{\alpha_3(\alpha_1 - \rho_1)(-\alpha_2 \rho_1 + \alpha_1 \rho_2)}{\alpha_1^2 \alpha_2 k \rho_1}, \frac{\alpha_3(\rho_1 - \alpha_1)}{\alpha_1 \alpha_2 k}, \frac{\rho_1 - \alpha_1}{\alpha_1 k}, 0\right)$
- $\left(0, \frac{\alpha_3(\rho_2 - \alpha_2)}{a_2^2 k}, \frac{\rho_2 - \alpha_2}{a_2 k}, 0\right)$
- $\left(0, \frac{-\alpha_2 \alpha_3 \gamma k L_{\text{máx}} + \alpha_2 \alpha_3 \rho_L + \alpha_2 \alpha_3 \rho_L + \alpha_2 \alpha_3 k L_{\text{máx}} \rho_L - \alpha_3 \rho_2 \rho_L}{a_2^2 k (\gamma k L_{\text{máx}} - \rho_L)}, \frac{-\alpha_2 \gamma k L_{\text{máx}} + \alpha_2 \rho_L + \alpha_2 k L_{\text{máx}} \rho_L - \rho_2 \rho_L}{\alpha_2 k (\gamma k L_{\text{máx}} - \rho_L)}, \frac{\gamma L_{\text{máx}} \rho_2 - \alpha_2 L_{\text{máx}} \rho_L}{\alpha_2 (\gamma k L_{\text{máx}} - \rho_L)}\right)$
- $\left(\frac{-\alpha_3(\alpha_2 \rho_1 - \alpha_1 \rho_2)(\alpha_1 \gamma k L_{\text{máx}} - \alpha_1 \rho_L - \alpha_1 k L_{\text{máx}} \rho_L + \rho_1 \rho_L)}{\alpha_1^2 \alpha_2 k \rho_1 (\gamma k L_{\text{máx}} - \rho_L)}, \frac{-\alpha_1 \alpha_3 \gamma k L_{\text{máx}} + \alpha_1 \alpha_3 \rho_L + \alpha_1 \alpha_3 k L_{\text{máx}} \rho_L - \alpha_3 \rho_1 \rho_L}{\alpha_1 \alpha_2 k (\gamma k L_{\text{máx}} - \rho_L)}, \frac{-\alpha_1 \gamma k L_{\text{máx}} + \alpha_1 \rho_L + \alpha_1 k L_{\text{máx}} \rho_L - \rho_1 \rho_L}{\alpha_1 k (\gamma k L_{\text{máx}} - \rho_L)}, \frac{\gamma L_{\text{máx}} \rho_1 - \alpha_1 L_{\text{máx}} \rho_L}{\alpha_1 (\gamma k L_{\text{máx}} - \rho_L)}\right)$

Para estudiar la estabilidad de los puntos hallados, calculamos la matriz de las derivadas relacionada con el sistema:

6.2 Estudio de las soluciones de equilibrio y estabilidad

$$\begin{pmatrix} -\alpha_1 + \frac{\rho_1}{1+(C_3+L)k} & 0 & \frac{-C_1 k \rho_1}{(1+(C_3+L)k)^2} & \frac{-C_1 k \rho_1}{(1+(C_3+L)k)^2} \\ \alpha_1 & -\alpha_2 + \frac{\rho_2}{1+(C_3+L)k} & \frac{-C_2 k \rho_2}{(1+(C_3+L)k)^2} & \frac{-C_2 k \rho_2}{(1+(C_3+L)k)^2} \\ 0 & \alpha_2 & -\alpha_3 & 0 \\ 0 & 0 & \frac{-C_L k (1 - \frac{L}{L_{\text{máx}}})}{(1+C_3 k)^2} & -\gamma + \frac{(1 - \frac{L}{L_{\text{máx}}}) \rho_1}{1+C_3 k} - \frac{L \rho_1}{(1+C_3 k) L_{\text{máx}}} \end{pmatrix} \quad (6.3)$$

Por la dificultad que entraña el tratamiento de los resultados de forma teórica, se realiza un estudio de estabilidad numérico. Los valores de los parámetros se presentan en la Tabla 6.1. Estos valores han sido tomados del desarrollo de los compartimentos en médula sana y se han estimado los parámetros del clon leucémico.

Parámetro	Modelo I	Modelo II
ρ_1	0.6931	0.6931
ρ_2	0.4621	0.4621
α_1	0.168	0.168
α_2	0.144	0.144
α_3	0.288	0.288
k	10^{-10}	10^{-10}
γ	0.000288	0.000288
$L_{\text{máx}}$	10^{12}	10^{12}

Tabla 6.1: Parámetros. Valores de los parámetros utilizados en cada modelo propuesto para el desarrollo de un clon leucémico en médula. El parámetro $L_{\text{máx}}$ se mide en número de células mientras que el resto de parámetros presentados tienen unidad dia^{-1} .

Debido a la complejidad del modelo y sus soluciones, además del número de parámetros asociado, el tratamiento que se realiza es un estudio numérico, como indicábamos, y es por ello que, para la estimación de los parámetros nos basamos en la bibliografía y no podemos realizar una estimación a través de las posibles soluciones en un modo simbólico. El estudio simbólico correspondiente a los modelos de médula sana presentados en la Sección 4.2 se pueden consultar en [10]. Para nuestro modelo, encontramos similitudes en algunos comportamientos y por tanto, tomamos de esta referencia los parámetros utilizados.

6. PRESENTACIÓN Y ESTUDIO DEL MODELO

Por todo lo anteriormente explicado, se tendrá en cuenta que las soluciones de equilibrio que se puedan obtener a continuación, carecerán de sentido biológico para valores negativos de alguno de los compartimentos, ya que corresponderían a poblaciones celulares negativas.

Modelo I

Se realiza el estudio para los casos en los que $p_L = p_1$ consiguiendo resultados similares para el caso $p_L = p_2$. Esto es, estamos considerando el caso en el que el clon aparece en el compartimento Pro-B, y adquiere las características proliferativas que se consideran para las células de este compartimento. Así, se obtienen las siguientes soluciones de equilibrio para (C_1, C_2, C_3, L) :

$$P_1 (0, -2.039 \times 10^{12}, -1.01987 \times 10^{12}, 1.04196 \times 10^{12})$$

Inestable. Son valores negativos con lo cual, carece de sentido biológico.

$$P_2 (0, 0, 0, 0)$$

Inestable. En el momento en el que aumentase alguno de los compartimentos, se modificarían todos.

$$P_3 (0, 0, 0, 9.99585 \times 10^{11})$$

Estable. Esto es lo que ocurriría si se dejase sin tratamiento. Las células leucémicas ocuparían la capacidad y acabarían con las células sanas.

$$P_4 (0, 4.41803 \times 10^{10}, 2.20901 \times 10^{10}, 0)$$

Inestable. En cuanto aparezca el clon, crecerá a costa de las sanas.

$$P_5 (-3.84878 \times 10^{11}, -2.02061 \times 10^{12}, -1.0103 \times 10^{12}, 1.04156 \times 10^{12})$$

Inestable. Valores negativos, carecen de sentido biológico.

$$P_6 (1.19081 \times 10^{10}, 6.25175 \times 10^{10}, 3.12588 \times 10^{10}, 0)$$

Inestable. Caso en el que no existiesen las leucémicas. Se comportaría de forma análoga a una médula sana. En cuanto aparezca el clon, se modificarían todos los compartimentos.

Modelo II

De la misma forma que para el Modelo I, se realiza el estudio para los casos en los que $p_L = p_1$ y obteniendo valores similares para el caso $p_L = p_2$. Es decir, volvemos a considerar el clon con capacidades celulares iguales a las células del compartimento Pro-B. Obtenemos en este caso las siguientes soluciones de equilibrio para (C_1, C_2, C_3, L) :

Se obtienen las siguientes soluciones de equilibrio para (C_1, C_2, C_3, L) :

$$P_1 (0, -6.79973 \times 10^{11}, -3.39957 \times 10^{11}, 1.04196 \times 10^{12})$$

Inestable. Son valores negativos con lo cual, carece de sentido biológico.

$$P_2 (0, 0, 0, 0)$$

Inestable. En el momento en el que aumentase alguno de los compartimentos, se modificarían todos.

$$P_3 (0, 0, 0, 9.99585 \times 10^{11})$$

Estable. Esto es lo que ocurriría si se dejase sin tratamiento. Las células leucémicas ocuparían la capacidad y acabarían con las células sanas.

$$P_4 (0, 1.47268 \times 10^{10}, .36338 \times 10^9, 0)$$

Inestable. En cuanto aparezca el clon, crecerá a costa de las sanas.

$$P_5 (-1.13837 \times 10^{11}, -5.97644 \times 10^{11}, -2.98822 \times 10^{11}, 1.04156 \times 10^{12})$$

Inestable. Valores negativos, carecen de sentido biológico.

$$P_6 (3.5221 \times 10^9, 1.84911 \times 10^{10}, 9.24555 \times 10^9, 0)$$

Inestable. Caso en el que no existiesen las leucémicas. Se comportaría de forma análoga a una médula sana. En cuanto aparezca el clon, se modificarían todos los compartimentos.

6.3 Simulaciones para el modelo propuesto

En el modelo propuesto para médula con clon leucémico tenemos que tener en cuenta que los valores iniciales para cada compartimento se corresponderán con los obtenidos como soluciones de equilibrio para la médula sana [10], mostrados en la Tabla 6.2. Se adjunta el estudio de la estabilidad de los modelos en médula sana en el software *Mathematica* en el

6. PRESENTACIÓN Y ESTUDIO DEL MODELO

Anexo B.1. Los parámetros usados para las próximas simulaciones, son los recogidos en la Tabla 6.1. Incluimos además, un valor inicial de células leucémicas $L_0 = 1$, es decir, se está reproduciendo el comienzo de la enfermedad, cuando una célula se vuelve cancerígena.

Dato Inicial	Modelo I	Modelo II
$C_1(0)$	7.1565×10^9	1.55331×10^9
$C_2(0)$	8.70833×10^{10}	1.89013×10^{10}
$C_3(0)$	2.6125×10^{10}	5.67039×10^9
$L(0)$	1	1

Tabla 6.2: Valores iniciales. Valores utilizados para la inicialización de cada modelo de clon leucémico en médula obtenidos de los valores de estabilidad de los modelos de médula sana. Resultados (4.15) y (4.16).

A continuación se muestran las simulaciones para cada modelo, Figura 6.1 para el Modelo I y Figura 6.2 para el Modelo II, adjuntándose el código implementado en *Python* en el Anexo B.4. Se representan las simulaciones mostrando para cada tiempo t , medido en días, las proporciones en médula, medidas en %, en las que se presentan cada uno de los compartimentos.

6.3 Simulaciones para el modelo propuesto

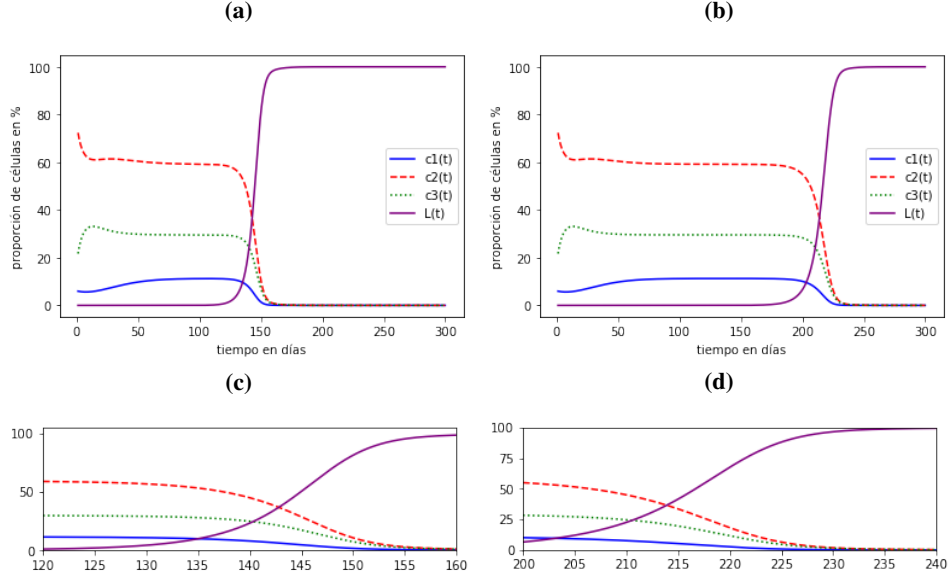


Figura 6.1: Dinámica evolutiva de los compartimentos celulares en presencia del clon leucémico (Modelo I). Valores iniciales (Tabla 6.2) $C_1(0) = 7.1565 \times 10^9$, $C_2(0) = 8.70833 \times 10^{10}$, $C_3(0) = 2.6125 \times 10^{10}$, $L(0) = 1$ asociados a los equilibrios en médula sana para el caso en el que sólo las células de Transición influyen en la señal de retroalimentación. Simulaciones realizadas para parámetros (Tabla 6.1) $\rho_1 = 0.6931$, $\rho_2 = 0.4621$, $\alpha_1 = 0.168$, $\alpha_2 = 0.144$, $\alpha_3 = 0.288$, $k = 10^{-10}$, $\gamma = 0.000288$, $L_{\text{máx}} = 10^{12}$. **(a)** Modelo I para el caso $\rho_L = \rho_1 = 0.6931$. La dinámica cambia bruscamente antes del día 150. **(b)** Modelo I para el caso $\rho_L = \rho_2 = 0.4621$. La dinámica del modelo cambia a partir del día 200. **(c)** Ampliación de la gráfica presentada en (a). Las células leucémicas (línea morada y continua) alcanzan mayor proporción en médula que las células C_1 (línea azul y continua) a partir del día 135. **(d)** Ampliación de la gráfica presentada en (b). En este caso, la proporción de células leucémicas, supera a la proporción de células C_1 a partir del día 202.

6. PRESENTACIÓN Y ESTUDIO DEL MODELO

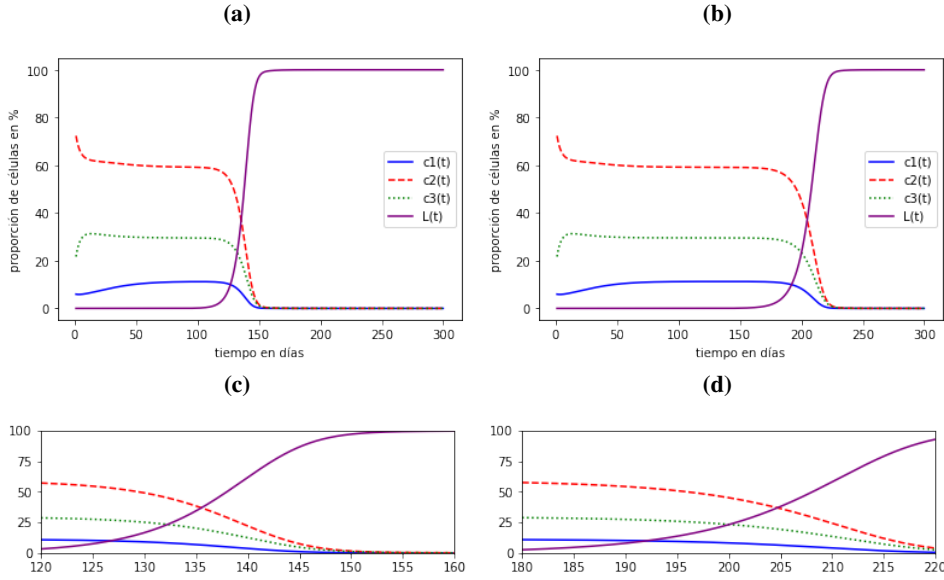


Figura 6.2: Dinámica evolutiva de los compartimentos celulares en presencia del clon leucémico (Modelo II). Valores iniciales (Tabla 6.2) $C_1(0) = 1.55331 \times 10^9$, $C_2(0) = 1.89013 \times 10^{10}$, $C_3(0) = 5.67039 \times 10^9$, $L(0) = 1$, asociados a los equilibrios en médula sana para el caso en el que sólo las células de Transición influyen en la señal de retroalimentación. Simulaciones realizadas para parámetros (Tabla 6.1) $\rho_1 = 0.6931$, $\rho_2 = 0.4621$, $\alpha_1 = 0.168$, $\alpha_2 = 0.144$, $\alpha_3 = 0.288$, $k = 10^{-10}$, $\gamma = 0.000288$, $L_{\text{máx}} = 10^{12}$. (a) Modelo II para el caso $\rho_L = \rho_1 = 0.6931$. La dinámica cambia bruscamente antes del día 150. (b) Modelo II para el caso $\rho_L = \rho_2 = 0.4621$. La dinámica del modelo cambia a partir del día 200. (c) Ampliación de la gráfica presentada en (a). Las células leucémicas (línea morada y continua) alcanzan mayor proporción en médula que las células C_1 (línea azul y continua) a partir del día 127. (d) Ampliación de la gráfica presentada en (b). En este caso, la proporción de células leucémicas, supera a la proporción de células C_1 a partir del día 192.

6.4 Interpretación de los resultados

El modelo describe el desarrollo de los compartimentos celulares a partir de la mutación de una célula por un fallo en el proceso de producción. En un estado inicial, la médula se encuentra en valores estables de los modelos de médula sana y se simula la aparición de una célula cancerígena.

La célula leucémica va reproducirse como se muestra en el modelo (6.1) con una tasa de proliferación, que según el compartimento en el que se encuentre, será más o menos acelerada.

Cuando se supone una proliferación $\rho_L = \rho_1$, para el Modelo I se tiene que la proporción de células leucémicas es igual a la proporción de células Pro-B entorno al día 135; igual a la proporción de células de Transición en el día 140; y supera a las células Pre-B en el día 143. En cambio, para el Modelo II, las células leucémicas alcanzan la misma proporción en médula que las células Pro-B entorno al día 127; igualan a las células de transición alrededor del día 132; y a las células Pre-B, pasado el día 135. Estos resultados se relacionan directamente con la señalización en las células leucémicas, ya que para la señal en la que solo influye las células de Transición, las leucémicas tardan más en reproducirse e invadir que en el modelo en el que influyen todas las células en la señalización. Ocurre de forma análoga para las simulaciones en los Modelos donde se asume $\rho_L = \rho_2$.

Además de la diferencia entre ambos modelos respecto a una misma tasa de proliferación tomada para ρ_L , cabe destacar que los casos donde la célula parte del compartimento Pre-B, esto es, donde ρ_L toma el valor ρ_2 , la invasión de las células leucémicas a la médula sana se produce de una forma más realentizada, tanto en el Modelo I como en el Modelo II. Esto es debido a que $\rho_2 < \rho_1$ y de ahí la variación en el tiempo transcurrido hasta que se produce la invasión total.

Como podemos comprobar, de manera general, ambos modelos se comportan de la misma forma. En todas las simulaciones del modelo con células leucémicas, podemos observar que las células sanas en presencia del clon coexisten durante un período de tiempo, que varía según el modelo y el caso, hasta que las células leucémicas ocupan la capacidad total en la médula no dejando espacio a las células sanas.

Respecto al modelo y las simulaciones, habrá que tener en cuenta que se está suponiendo una evolución de la enfermedad sin tratamiento que reduzca la proporción de células leucémicas en ningún momento y por tanto explica que desde el comienzo de la enfermedad hasta aproximadamente 3 meses después, no hay ningún cambio brusco en las proporciones respecto a la médula sana.

Estas simulaciones dan pie a estudios relacionados con la aplicación del tratamiento y en qué tiempos sería conveniente realizar una disminución de las células leucémicas para la

6. PRESENTACIÓN Y ESTUDIO DEL MODELO

coexistencia de todos los compartimentos celulares hasta la eliminación de todas las células cancerígenas.

Resultados y Conclusiones

A continuación detallaremos los resultados y conclusiones de este estudio. Se mostrará la aplicabilidad de los resultados obtenidos y las mejoras que se pueden llevar a cabo en los distintos procesos estudiados.

7.1 Aplicaciones de los resultados

Como se mostró en la Sección 5.5, hemos utilizado técnicas de *clusterización* para obtener la proporción en la que se presentan los distintos compartimentos celulares en el proceso de linfopoyesis. De esta forma y junto con los valores de equilibrio en médula sana, hemos desarrollado un método de detección de fallos en estos compartimentos para poder definir las propiedades que adquiere el clon leucémico presente.

Esto permite la determinación de los parámetros para el nuevo modelo presentado y facilita la representación de la evolución de las células en el proceso de linfopoyesis en presencia de un clon leucémico.

Por otro lado, el algoritmo utilizado para la clasificación de los pacientes según su diagnóstico ha llegado a determinar de forma correcta el 76.31 % del total. Esto es un primer paso importante para automatizar la clasificación de datos en médula. Hasta ahora, la

7. RESULTADOS Y CONCLUSIONES

clasificación de células encontrada en la literatura consiste en clusterizar células de muestras de sangre. En este trabajo se realiza la automatización de la clasificación de células en la linfopoyesis, proceso en el que las células se están diferenciando, mientras que en la sangre, las células ya son maduras.

Para la mejora del método presentado, se requieren unos marcadores de las células que no vienen reflejados en nuestros datos y que permitirán diferenciar los tipos de leucemia correctamente en los casos en los que existe un diagnóstico erróneo. Una vez mejorado el algoritmo, podremos contrastar el algoritmo propuesto con un mayor número de pacientes para verificarlos y facilitar estudios futuros.

Para la validación del modelo presentado en el Capítulo 6 y la comprobación de los parámetros usados, se verá necesario analizar nuevas muestras de pacientes. Según lo reflejado en las simulaciones y los resultados que se deducen, sería posible determinar el tiempo que ha transcurrido desde el inicio de la enfermedad, siempre y cuando tengamos un seguimiento del paciente en dos momentos distintos sin haber aplicado tratamiento. Así, se reproduciría la evolución de los distintos compartimentos celulares de una manera más precisa y ajustada al paciente en cuestión. En los datos reales hasta ahora tomados, no existe este tipo de comparación, ya que una vez diagnosticada la Leucemia en un paciente, pasa a ser tratado según los protocolos de actuación.

El modelo propuesto podrá aplicarse a estudios más elaborados que modelicen el mismo fenómeno, concretando para cada grupo de pacientes ciertos valores y diferenciando entre los distintos tipos de casos que pueden darse. A su vez, la aplicación de este modelo a datos tomados de muestras de pacientes, atendiendo a los marcadores celulares necesarios, permitirá una mejora del propio modelo.

Todas estas aplicaciones pueden ser llevadas a clínica para analizar los distintos resultados teóricos y profundizar en la realidad de los modelos. Esto evidencia que la Oncología Matemática se trata de una vía de estudio y una nueva herramienta para la mejora del análisis de distintos procesos biológicos.

7.2 Conclusiones

En este estudio se ha trabajado sobre los siguientes aspectos:

Hemos realizado una introducción a los conceptos biológicos fundamentales para basar nuestro trabajo en las principales referencias en cuanto a cáncer y leucemia. Hemos analizado y sintetizado la bibliografía presentada para una mayor comprensión de los procesos de diferenciación celular y especialización de linfocitos tipo B.

Hemos plasmado los conceptos teóricos relacionados con el análisis de ecuaciones diferenciales para su aplicación en nuestro estudio y hemos incluido los resultados principales que nos facilitan la determinación de la estabilidad de los modelos.

Hemos realizado una revisión bibliográfica sobre los modelos de crecimiento de poblaciones celulares, mostrando en este trabajo los estudios principales más influyentes para nuestra nueva propuesta. Nos basamos en los modelos de hematopoyesis [33] y en los modelos de linfopoyesis en médula sana [10]. De estos estudios, hemos reproducido las simulaciones y hemos comprobado los resultados obtenidos para realizar un análisis análogo de los nuevos modelos propuestos.

Hemos analizado datos de pacientes reales con técnicas de *clusterización* para la clasificación automática de los mismos. De esta forma hemos desarrollado en *Python* un algoritmo que nos permite estudiar automáticamente la composición de una muestra de sangre y su comparación con los niveles en médula sana. Para ello, hemos recurrido a una diferenciación de los tres compartimentos celulares a través de sus marcadores. Lo cual, nos ha llevado a la comprensión del funcionamiento de un citómetro de flujo. Por otro lado, y con un tratamiento distinto de los datos, hemos clasificado el grupo de pacientes según su diagnóstico. Se han obtenido resultados que se asemejan a las muestras de médula disponibles en nuestro estudio.

A partir de los datos de los pacientes, hemos obtenido información sobre el comportamiento de la leucemia y hemos propuesto un modelo de diferenciación de linfocitos tipo B en presencia de clon leucémico a partir de los modelos en médula sana y el estudio

7. RESULTADOS Y CONCLUSIONES

del comportamiento de las células leucémicas. Hemos realizado el estudio de los modelos presentados y hemos simulado las distintas situaciones obteniendo resultados aplicables a futuros estudios.

Todos estos resultados, han sido fruto de la aplicación de los conocimientos adquiridos en el Máster, tanto a nivel teórico para la Modelización de los problemas y su contraste con la realidad, como el uso de los *softwares Python* y *Mathematica* y el uso de paquetes como *TikZ* para la creación y presentación de imágenes en el editor de textos *L^AT_EX*.

Lo señalado anteriormente, nos lleva a proponer distintos proyectos futuros para profundizar en este estudio con el fin de continuar ampliando conocimientos y utilizar nuevas herramientas.

Ampliación y Proyectos Futuros

Una mejora del modelo y de los algoritmos propuestos implicaría un resultado influyente en futuros proyectos. Realizando un modelo que se ajuste a la aparición de un clon leucémico en una médula sana, de manera contrastada, podremos establecer nuevos protocolos para el tratamiento de la enfermedad.

Pretendemos evitar la recaída de los pacientes con la aportación de nuevos estudios en este ámbito. La investigación y la resolución de estos problemas matemáticos se irán analizando y presentando en proyectos futuros.

8.1 Planteamiento de proyecto futuro

Una vez validado el modelo y estudiados los diagnósticos de cada paciente, podremos analizar y aproximar en cada caso en qué momento se encuentra desde que se originó la leucemia. Esto solo es posible si tenemos la información suficiente.

A día de hoy, los datos de los que disponemos son muestras de las que no sabemos en qué momento de la enfermedad fueron tomadas, con lo cual no sabemos su evolución ni podemos aproximar el tiempo que ha transcurrido. Se podría hacer un estudio más exhaustivo y un refinamiento de los parámetros en el momento en el que tengamos información del porcentaje de blastos en sangre que tengan los pacientes y un seguimiento en varios tiempos para cada paciente para poder estudiar la evolución del clon.

8. AMPLIACIÓN Y PROYECTOS FUTUROS

Con los resultados obtenidos a lo largo de este trabajo de fin de máster, los proyectos planteados para la continuación de la investigación en los modelos de LLA, se enumeran como sigue:

- Validar el algoritmo de clasificación de tipos de Leucemia y su uso para predecir el diagnóstico de un paciente.
- Validar y/o ajustar los modelos propuestos a través de un mayor número de datos reales de las que destacar los factores que se han indicado necesarios a lo largo del estudio.
- Proponer modelos donde el clon aparece en múltiples estados madurativos y toma distintas propiedades en su evolución como se sugiere en [23].
- Estudiar los modelos de linfopoyesis con la aparición de un clon leucémico y bajo condiciones de tratamiento.
- Aproximar el inicio de la enfermedad a partir de la proporción de blastos en sangre de un paciente, para poder aplicar un tratamiento más efectivo y estudiar la evolución de la enfermedad con tratamiento.
- Plantear modelos que incluyan las dosis del tratamiento y asumir un retardo entre la aplicación del tratamiento y su efecto.

A partir de la obtención de los datos requeridos para su posterior tratamiento y un avance en el estudio propuesto, se permitirá una detección precoz de la enfermedad y caracterizarla para una mejora en los protocolos de actuación.

ANEXO



Abreviaturas

Se añade en este apartado una guía para el lector en la que se recogen las abreviaturas que se han incluido a lo largo del trabajo.

- **CD** Cluster of Differentiation
- **FCS** Flow Cytometry Standard
- **IPT** Inmunofenotipo
- **LLA** Leucemia Linfoblástica Aguda
- **LLA-B** Leucemia Linfoblástica Aguda en la línea de linfocitos tipo B

ANEXO

B

Estabilidad y simulaciones de los modelos

En este apéndice se incluyen el estudio de la estabilidad tanto para médula sana como para modelo con clon, realizado con el software *Mathematica* versión 9.0. Se añaden también las simulaciones de los modelos realizadas en *Python 3.7.3*.

B.1 Estudio de la estabilidad en los modelos de médula sana

ESTUDIO DE ESTABILIDAD EN MODELOS DE MÉDULA SANA

PARÁMETROS USADOS

```
Quit[];

p1 = 0.0289;
p2 = 0.0193;
a1 = 0.008;
a2 = 0.006;
a3 = 0.02;
k = 10-10;
```

Modelo A.1 : Solo las células maduras influyen en la señal.

```
Quit[];

f1[C1_, C2_, C3_] :=  $\frac{1}{1 + k (C3)}$  p1 * C1 - a1 * C1;
f2[C1_, C2_, C3_] :=  $\frac{1}{1 + k (C3)}$  p2 * C2 + a1 * C1 - a2 * C2;
f3[C1_, C2_, C3_] := a2 C2 - a3 C3;
```

Equilibrio y estabilidad

```
Equilibrios = Solve[
  {f1[C1, C2, C3] == 0 && f2[C1, C2, C3] == 0 && f3[C1, C2, C3] == 0}, {C1, C2, C3} ];
J[C1_, C2_, C3_] := Evaluate[{
  {D[f1[C1, C2, C3], C1], D[f1[C1, C2, C3], C2], D[f1[C1, C2, C3], C3]},
  {D[f2[C1, C2, C3], C1], D[f2[C1, C2, C3], C2], D[f2[C1, C2, C3], C3]},
  {D[f3[C1, C2, C3], C1], D[f3[C1, C2, C3], C2], D[f3[C1, C2, C3], C3]}}];

Definition[J];

For[i = 1, i ≤ Length[Equilibrios], i++,
  peq = Equilibrios[[i]];
  J0 = J[peq[[1, 2]], peq[[2, 2]], peq[[3, 2]]];
  If[Det[J0] == 0, Print["En el punto de equilibrio",
    peq, "no puede aplicarse H-G porque el determinante es 0"],

    If[Re[Eigenvalues[J0][[1]]] < 0 &&
      Re[Eigenvalues[J0][[2]]] < 0 && Re[Eigenvalues[J0][[3]]] < 0,
      Print["La solución de equilibrio ", peq, " es estable"],
      Print["La solución de equilibrio ", peq, " es inestable"]]
  ]
]
```

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 0., C3 \rightarrow 0.\}$ es inestable

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 7.38889 \times 10^{10}, C3 \rightarrow 2.21667 \times 10^{10}\}$ es inestable

La solución de equilibrio

$\{C1 \rightarrow 7.1565 \times 10^9, C2 \rightarrow 8.70833 \times 10^{10}, C3 \rightarrow 2.6125 \times 10^{10}\}$ es estable

Modelo A.2 : Todas las células influyen en la señal.

$$f1[C1_ , C2_ , C3_] := \frac{1}{1 + k (C1 + C2 + C3)} p1 * C1 - a1 * C1;$$

$$f2[C1_ , C2_ , C3_] := \frac{1}{1 + k (C1 + C2 + C3)} p2 * C2 + a1 * C1 - a2 * C2;$$

$$f3[C1_ , C2_ , C3_] := a2 C2 - a3 C3;$$

Equilibrio y estabilidad

```
Equilibrios = Solve[
  {f1[C1, C2, C3] == 0 && f2[C1, C2, C3] == 0 && f3[C1, C2, C3] == 0}, {C1, C2, C3}];
J[C1_, C2_, C3_] := Evaluate[{
  {D[f1[C1, C2, C3], C1], D[f1[C1, C2, C3], C2], D[f1[C1, C2, C3], C3]},
  {D[f2[C1, C2, C3], C1], D[f2[C1, C2, C3], C2], D[f2[C1, C2, C3], C3]},
  {D[f3[C1, C2, C3], C1], D[f3[C1, C2, C3], C2], D[f3[C1, C2, C3], C3]}}];

Definition[J];

For[i = 1, i <= Length[Equilibrios], i++,
  peq = Equilibrios[[i]];
  J0 = J[peq[[1, 2]], peq[[2, 2]], peq[[3, 2]]];
  If[Det[J0] == 0, Print["En el punto de equilibrio",
    peq, "no puede aplicarse H-G porque el determinante es 0"],

    If[Re[Eigenvalues[J0][[1]]] < 0 &&
      Re[Eigenvalues[J0][[2]]] < 0 && Re[Eigenvalues[J0][[3]]] < 0,
      Print["La solución de equilibrio ", peq, " es estable"],
      Print["La solución de equilibrio ", peq, " es inestable"]]
  ]
]
```

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 0., C3 \rightarrow 0.\}$ es inestable

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 1.70513 \times 10^{10}, C3 \rightarrow 5.11538 \times 10^9\}$ es inestable

La solución de equilibrio

$\{C1 \rightarrow 1.55331 \times 10^9, C2 \rightarrow 1.89013 \times 10^{10}, C3 \rightarrow 5.67039 \times 10^9\}$ es estable

B. ESTABILIDAD Y SIMULACIONES DE LOS MODELOS

B.2 Simulaciones de modelos de médula sana

Códigos asociados a la representación de los modelos de médula sana presentados en la Sección 4.2.

Simulaciones para modelos tipo A

```
[1]: %matplotlib inline
import numpy as np
import pylab as p
import matplotlib.pyplot as plt
import scipy.integrate as sp
from scipy.integrate import solve_ivp
```

Parámetros usados

```
[5]: p1 = 0.0289;
p2 = 0.0193;
a1 = 0.008;
a2 = 0.006;
a3 = 0.02;
k = 10**(-10);
C0=[10**6,0,0];
```

Modelo A.1. Solo las células tardías influyen en la señal.

```
[3]: def modelo1(C,t):
    c1=C[0]
    c2=C[1]
    c3=C[2]
    dc1dt=p1*c1/(1+k*c3)-a1*c1
    dc2dt=p2*c2/(1+k*c3) +a1*c1-a2*c2
    dc3dt=a2*c2-a3*c3
    dCdt=[dc1dt,dc2dt,dc3dt]
    return dCdt
```

```
[4]: #en el intervalo (1,3000), quiero 6000 puntos:

t=np.linspace(1,3000,6000)

# almacenaje de las soluciones
c1 = np.empty_like(t)
c2 = np.empty_like(t)
```

```

c3 = np.empty_like(t)

# Condiciones iniciales
c1[0] = C0[0]
c2[0] = C0[1]
c3[0] = C0[2]

# Solución EDOs
#tomo desde 1 a 6000, ie, todos los puntos que he definido

for i in range(1,6000):

    tspan = [t[i-1],t[i]]

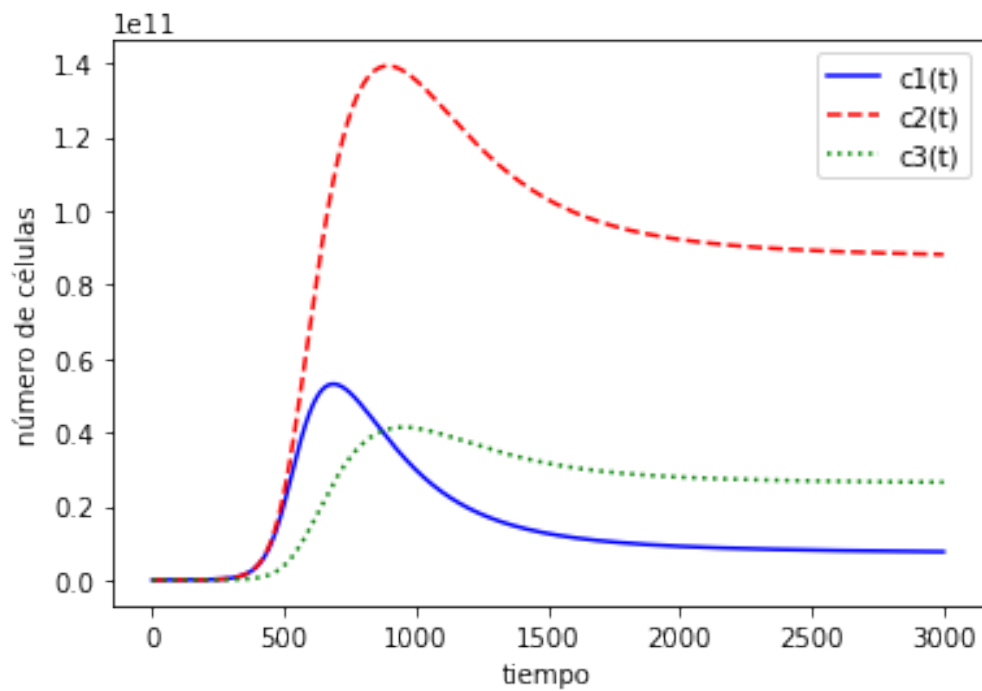
    C = sp.odeint(modelo1,C0,tspan)

    #Almacenamos los datos para dibujarlos
    c1[i] = C[1][0]
    c2[i] = C[1][1]
    c3[i] = C[1][2]
    #Próxima condición inicial
    C0 = C[1]

# Gráfica

plt.plot(t,c1,'b-',label='c1(t)')
plt.plot(t,c2,'r--',label='c2(t)')
plt.plot(t,c3,'g',linestyle='dotted',label='c3(t)')
plt.ylabel('número de células')
plt.xlabel('tiempo')
plt.legend(loc='best')
mod1graf=plt.show()

```



Modelo A.2. Todas las células tardías influyen en la señal.

```
[6]: def modelo2(C,t):
    c1=C[0]
    c2=C[1]
    c3=C[2]
    dc1dt=p1*c1/(1+k*(c1+c2+c3))-a1*c1
    dc2dt=p2*c2/(1+k*(c1+c2+c3)) +a1*c1-a2*c2
    dc3dt=a2*c2-a3*c3
    dCdt=[dc1dt,dc2dt,dc3dt]
    return dCdt
```

```
[7]: #en el intervalo (1,3000), quiero 6000 puntos:

t=np.linspace(1,3000,6000)

# almacenaje de las soluciones
c1 = np.empty_like(t)
c2 = np.empty_like(t)
c3 = np.empty_like(t)

# Condiciones iniciales
c1[0] = C0[0]
```

```

c2[0] = C0[1]
c3[0] = C0[2]

# Solución EDOs
#tomo desde 1 a 6000, ie, todos los puntos que he definido

for i in range(1,6000):

    tspan = [t[i-1],t[i]]

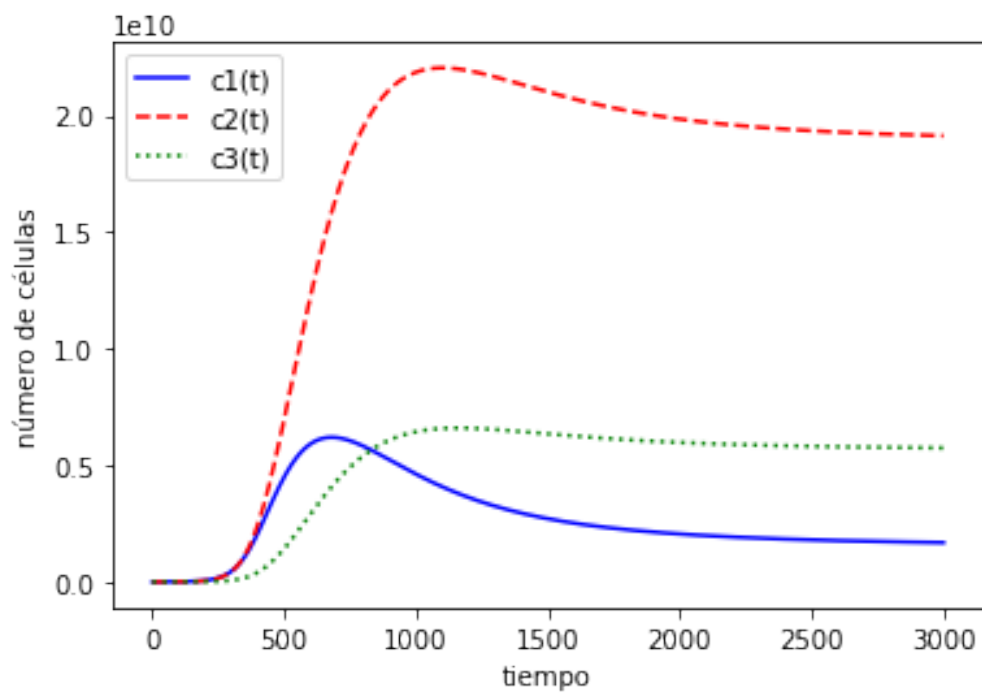
    C = sp.odeint(modelo2,C0,tspan)

    #Almacenamos los datos para dibujarlos
    c1[i] = C[1][0]
    c2[i] = C[1][1]
    c3[i] = C[1][2]
    #Próxima condición inicial
    C0 = C[1]

# Gráfica

plt.plot(t,c1,'b-',label='c1(t)')
plt.plot(t,c2,'r--',label='c2(t)')
plt.plot(t,c3,'g',linestyle='dotted',label='c3(t)')
plt.ylabel('número de células')
plt.xlabel('tiempo')
plt.legend(loc='best')
plt.show()

```

[]:

B. ESTABILIDAD Y SIMULACIONES DE LOS MODELOS

B.3 Estudio de la estabilidad en los modelos de médula con clon leucémico

ESTUDIO TEÓRICO DE ESTABILIDAD DEL MODELO DE LEUCÉMICAS

del equilibrio y la estabilidad por el Teorema de HG.

En los parámetros podemos considerar $p_L=p_1$ o $p_L=p_2$.

PARÁMETROS

In[19]:= Quit[] ;

In[32]:= $p_L = p_1$; (* modelo en el que el clon se encuentra en el compartimento de células Pro-B *)

(* $p_L=p_2$ para el caso en el que el clon se desarrolla con características de células Pre-B*)

Modelo A.1. : Solo las células maduras influyen en la señal.

In[8]:=
$$f_1[C_1, C_2, C_3, CL] := \frac{1}{1 + k (CL + C_3)} p_1 * C_1 - a_1 * C_1;$$

$$f_2[C_1, C_2, C_3, CL] := \frac{1}{1 + k (CL + C_3)} p_2 * C_2 + a_1 * C_1 - a_2 * C_2;$$

$$f_3[C_1, C_2, C_3, CL] := a_2 C_2 - a_3 C_3;$$

$$f_4[C_1, C_2, C_3, CL] := \frac{p_L * CL}{1 + k (C_3)} * \left(1 - \frac{CL}{L_{max}}\right) - \gamma * CL;$$

Puntos de equilibrio

In[12]:=

Equilibrios = Solve[{f1[C1, C2, C3, CL] == 0 && f2[C1, C2, C3, CL] == 0 && f3[C1, C2, C3, CL] == 0 && f4[C1, C2, C3, CL] == 0}, {C1, C2, C3, CL}]

Out[12]=
$$\left\{ \left\{ C_1 \rightarrow 0, C_2 \rightarrow 0, C_3 \rightarrow 0, CL \rightarrow 0 \right\}, \left\{ C_1 \rightarrow 0, C_2 \rightarrow 0, C_3 \rightarrow 0, CL \rightarrow \frac{-\gamma L_{max} + L_{max} p_1}{p_1} \right\}, \right.$$

$$\left\{ C_1 \rightarrow \frac{a_3 (a_1 - p_1) (-a_2 p_1 + a_1 p_2)}{a_1^2 a_2 k p_1}, C_2 \rightarrow \frac{-a_1 a_3 + a_3 p_1}{a_1 a_2 k}, C_3 \rightarrow \frac{-a_1 + p_1}{a_1 k}, CL \rightarrow 0 \right\},$$

$$\left\{ C_1 \rightarrow 0, C_2 \rightarrow \frac{-a_2 a_3 + a_3 p_2}{a_2^2 k}, C_3 \rightarrow \frac{-a_2 + p_2}{a_2 k}, CL \rightarrow 0 \right\},$$

$$\left\{ C_1 \rightarrow 0, C_2 \rightarrow \frac{(-a_2 a_3 \gamma k L_{max} + a_2 a_3 p_1 + a_2 a_3 k L_{max} p_1 - a_3 p_1 p_2)}{(a_2^2 k (\gamma k L_{max} - p_1))}, \right.$$

$$C_3 \rightarrow \frac{-a_2 \gamma k L_{max} + a_2 p_1 + a_2 k L_{max} p_1 - p_1 p_2}{a_2 k (\gamma k L_{max} - p_1)}, CL \rightarrow \frac{-a_2 L_{max} p_1 + \gamma L_{max} p_2}{a_2 (\gamma k L_{max} - p_1)} \left. \right\},$$

$$\left\{ C_1 \rightarrow \frac{a_3 (a_1 \gamma k L_{max} - a_1 p_1 - a_1 k L_{max} p_1 + p_1^2) (-a_2 p_1 + a_1 p_2)}{(a_1^2 a_2 k (\gamma k L_{max} - p_1) p_1)}, \right.$$

$$C_2 \rightarrow \frac{-a_1 a_3 \gamma k L_{max} + a_1 a_3 p_1 + a_1 a_3 k L_{max} p_1 - a_3 p_1^2}{a_1 a_2 k (\gamma k L_{max} - p_1)},$$

$$C_3 \rightarrow \frac{-a_1 \gamma k L_{max} + a_1 p_1 + a_1 k L_{max} p_1 - p_1^2}{a_1 k (\gamma k L_{max} - p_1)}, CL \rightarrow \frac{-a_1 L_{max} p_1 + \gamma L_{max} p_1}{a_1 (\gamma k L_{max} - p_1)} \left. \right\}$$

In[13]:= Length[Equilibrios]

Out[13]= 6

```
In[14]:= peq1 = {Equilibrios[[1]][[1, 2]], Equilibrios[[1]][[2, 2]],
  Equilibrios[[1]][[3, 2]], Equilibrios[[1]][[4, 2]]};
peq2 = {Equilibrios[[2]][[1, 2]], Equilibrios[[2]][[2, 2]],
  Equilibrios[[2]][[3, 2]], Equilibrios[[2]][[4, 2]]};
peq3 = {Equilibrios[[3]][[1, 2]], Equilibrios[[3]][[2, 2]],
  Equilibrios[[3]][[3, 2]], Equilibrios[[3]][[4, 2]]};
peq4 = {Equilibrios[[4]][[1, 2]], Equilibrios[[4]][[2, 2]],
  Equilibrios[[4]][[3, 2]], Equilibrios[[4]][[4, 2]]};
peq5 = {Equilibrios[[5]][[1, 2]], Equilibrios[[5]][[2, 2]],
  Equilibrios[[5]][[3, 2]], Equilibrios[[5]][[4, 2]]};
peq6 = {Equilibrios[[6]][[1, 2]], Equilibrios[[6]][[2, 2]],
  Equilibrios[[6]][[3, 2]], Equilibrios[[6]][[4, 2]]};
```

Definición de la matriz Jacobiana

```
In[20]:= J[{C1_, C2_, C3_, CL_}] := Evaluate[{
  {D[f1[C1, C2, C3, CL], C1], D[f1[C1, C2, C3, CL], C2],
    D[f1[C1, C2, C3, CL], C3], D[f1[C1, C2, C3, CL], CL]},
  {D[f2[C1, C2, C3, CL], C1], D[f2[C1, C2, C3, CL], C2],
    D[f2[C1, C2, C3, CL], C3], D[f2[C1, C2, C3, CL], CL]},
  {D[f3[C1, C2, C3, CL], C1], D[f3[C1, C2, C3, CL], C2],
    D[f3[C1, C2, C3, CL], C3], D[f3[C1, C2, C3, CL], CL]},
  {D[f4[C1, C2, C3, CL], C1], D[f4[C1, C2, C3, CL], C2],
    D[f4[C1, C2, C3, CL], C3], D[f4[C1, C2, C3, CL], CL]}
}];
```

Definition[J];

```
In[22]:= MatrixForm[J[{C1, C2, C3, CL}]]
```

Out[22]//MatrixForm=

$$\begin{pmatrix} -a1 + \frac{p1}{1+(C3+CL)k} & 0 & -\frac{C1k p1}{(1+(C3+CL)k)^2} & -\frac{C1k p1}{(1+(C3+CL)k)^2} \\ a1 & -a2 + \frac{p2}{1+(C3+CL)k} & -\frac{C2k p2}{(1+(C3+CL)k)^2} & -\frac{C2k p2}{(1+(C3+CL)k)^2} \\ 0 & a2 & -a3 & 0 \\ 0 & 0 & -\frac{CLk \left(1 - \frac{CL}{Lmax}\right) p1}{(1+C3k)^2} & -\gamma + \frac{\left(1 - \frac{CL}{Lmax}\right) p1}{1+C3k} - \frac{CL p1}{(1+C3k) Lmax} \end{pmatrix}$$

Autovalores y condiciones para la estabilidad de cada punto

Ejemplo en Punto de equilibrio 1

```
In[23]:= autovalores1 = Eigenvalues[J[peq1]]
```

```
Out[23]= {-a3, -a1 + p1, -gamma + p1, -a2 + p2}
```

```
Condiciones1 = Reduce[Re[autovalores1[[1]]] < 0 &&
  Re[autovalores1[[2]]] < 0 && Re[autovalores1[[3]]] < 0 &&
  Re[autovalores1[[4]]] < 0, {a1, a2, a3, p1, p2, gamma}];
```

```
In[25]:= Condiciones1
```

```
Out[25]= Re[a3] > 0 && Re[p1] < Re[a1] && Re[p2] < Re[a2] && Re[gamma] > Re[p1]
```

Autovalores para los demás puntos

```
In[26]:= autovalores2 = Eigenvalues[J[peq2]];
autovalores3 = Eigenvalues[J[peq3]];
autovalores4 = Eigenvalues[J[peq4]];
autovalores5 = Eigenvalues[J[peq5]];
autovalores6 = Eigenvalues[J[peq6]];
```

Vemos que para el punto de equilibrio 2 ya se complica el estudio de la negatividad de los autovalores, existiendo muchos casos a los que atender.

```
In[31]:= autovalores2
```

```
Out[31]= { -a3, gamma - p1,  $\frac{a1 \text{ gamma } k \text{ Lmax} - a1 \text{ p1} - a1 \text{ k Lmax } p1 + p1^2}{- \text{ gamma } k \text{ Lmax} + p1 + k \text{ Lmax } p1}$ ,  
-  $\frac{a2 \text{ gamma } k \text{ Lmax} + a2 \text{ p1} + a2 \text{ k Lmax } p1 - p1 \text{ p2}}{- \text{ gamma } k \text{ Lmax} + p1 + k \text{ Lmax } p1}$  }
```

De hecho, **nuestros recursos no nos permiten compilar el código** que nos indica las condiciones que deben cumplir los parámetros para que los distintos puntos de equilibrio sean estables.

Condiciones para los demás puntos

```
Condiciones2 = Reduce[Re[autovalores2[[1]]] < 0 &&
  Re[autovalores2[[2]]] < 0 && Re[autovalores2[[3]]] < 0 &&
  Re[autovalores2[[4]]] < 0, {a1, a2, a3, p1, p2, gamma}];
Condiciones3 = Reduce[Re[autovalores3[[1]]] < 0 &&
  Re[autovalores3[[2]]] < 0 && Re[autovalores3[[3]]] < 0 &&
  Re[autovalores3[[4]]] < 0, {a1, a2, a3, p1, p2, gamma}];
Condiciones4 = Reduce[Re[autovalores4[[1]]] < 0 &&
  Re[autovalores4[[2]]] < 0 && Re[autovalores4[[3]]] < 0 &&
  Re[autovalores4[[4]]] < 0, {a1, a2, a3, p1, p2, gamma}];
Condiciones5 = Reduce[Re[autovalores5[[1]]] < 0 &&
  Re[autovalores5[[2]]] < 0 && Re[autovalores5[[3]]] < 0 &&
  Re[autovalores5[[4]]] < 0, {a1, a2, a3, p1, p2, gamma}];
Condiciones6 = Reduce[Re[autovalores6[[1]]] < 0 &&
  Re[autovalores6[[2]]] < 0 && Re[autovalores6[[3]]] < 0 &&
  Re[autovalores6[[4]]] < 0, {a1, a2, a3, p1, p2, gamma}];
```

ESTUDIO DE ESTABILIDAD DEL MODELO DE LEUCÉMICAS

Estudiamos el equilibrio y la estabilidad por el Teorema de HG.
En los parámetros podemos considerar $p_L=p_1$ o $p_L=p_2$.

PARÁMETROS (convertidos en horas)

```
Quit[ ];

ln[56]:= p1 = N[Log[2] / 1];
p2 = N[Log[2] / (1.5)];
a1 = 0.168;
a2 = 0.144;
a3 = 0.288;
k = 10-10;

(*Relacionados con el clon*)
gamma = 0.001 * a3;
Lmax = 1012;
pL = p1; (* modelo en el que el clon se
          encuentra en el compartimento de células Pro-B *)

(*pL=p2 para el caso en el que el clon
se desarrolla con características de células Pre-B*)
```

Modelo A.1. : Solo las células maduras influyen en la señal.

```
ln[48]:= f1[C1_, C2_, C3_, CL_] :=  $\frac{1}{1 + k (CL + C3)}$  p1 * C1 - a1 * C1;

f2[C1_, C2_, C3_, CL_] :=  $\frac{1}{1 + k (CL + C3)}$  p2 * C2 + a1 * C1 - a2 * C2;
f3[C1_, C2_, C3_, CL_] := a2 C2 - a3 C3;
f4[C1_, C2_, C3_, CL_] :=  $\frac{pL * CL}{1 + k (C3)}$  *  $\left(1 - \frac{CL}{Lmax}\right)$  - gamma * CL;
```

Equilibrio y estabilidad

```

In[52]:= Equilibrios = Solve[{f1[C1, C2, C3, CL] == 0 && f2[C1, C2, C3, CL] == 0 &&
    f3[C1, C2, C3, CL] == 0 && f4[C1, C2, C3, CL] == 0}, {C1, C2, C3, CL}];
J[C1_, C2_, C3_, CL_] := Evaluate[{
    {D[f1[C1, C2, C3, CL], C1], D[f1[C1, C2, C3, CL], C2],
    D[f1[C1, C2, C3, CL], C3], D[f1[C1, C2, C3, CL], CL]},

    {D[f2[C1, C2, C3, CL], C1], D[f2[C1, C2, C3, CL], C2],
    D[f2[C1, C2, C3, CL], C3], D[f2[C1, C2, C3, CL], CL]},

    {D[f3[C1, C2, C3, CL], C1], D[f3[C1, C2, C3, CL], C2],
    D[f3[C1, C2, C3, CL], C3], D[f3[C1, C2, C3, CL], CL]},

    {D[f4[C1, C2, C3, CL], C1], D[f4[C1, C2, C3, CL], C2],
    D[f4[C1, C2, C3, CL], C3], D[f4[C1, C2, C3, CL], CL]}
}];

Definition[J];

For[i = 1, i ≤ Length[Equilibrios], i++,
    peq = Equilibrios[[i]];
    J0 = J[peq[[1, 2]], peq[[2, 2]], peq[[3, 2]], peq[[4, 2]]];
    If[Det[J0] == 0, Print["En el punto de equilibrio",
        peq, "no puede aplicarse H-G porque el determinante es 0."],

        If[Re[Eigenvalues[J0][[1]]] < 0 && Re[Eigenvalues[J0][[2]]] < 0 &&
            Re[Eigenvalues[J0][[3]]] < 0 && Re[Eigenvalues[J0][[4]]] < 0,
            Print["La solución de equilibrio ", peq, " es estable."],
            Print["La solución de equilibrio ", peq, " es inestable."]]
    ]
]

```

La solución de equilibrio

$\{C1 \rightarrow 0., C2 \rightarrow -2.03974 \times 10^{12}, C3 \rightarrow -1.01987 \times 10^{12}, CL \rightarrow 1.04196 \times 10^{12}\}$ es inestable.

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 0., C3 \rightarrow 0., CL \rightarrow 0.\}$ es inestable.

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 0., C3 \rightarrow 0., CL \rightarrow 9.99585 \times 10^{11}\}$ es estable.

La solución de equilibrio

$\{C1 \rightarrow 0., C2 \rightarrow 4.41803 \times 10^{10}, C3 \rightarrow 2.20901 \times 10^{10}, CL \rightarrow 0.\}$ es inestable.

La solución de equilibrio $\{C1 \rightarrow -3.84878 \times 10^{11},$

$C2 \rightarrow -2.02061 \times 10^{12}, C3 \rightarrow -1.0103 \times 10^{12}, CL \rightarrow 1.04156 \times 10^{12}\}$ es inestable.

La solución de equilibrio

$\{C1 \rightarrow 1.19081 \times 10^{10}, C2 \rightarrow 6.25175 \times 10^{10}, C3 \rightarrow 3.12588 \times 10^{10}, CL \rightarrow 0.\}$ es inestable.

Modelo A.2. : Todas las células influyen en la señal.

`Quit[]`

```
In[65]:= f1[C1_, C2_, C3_, CL_] := 
$$\frac{1}{1 + k (CL + C1 + C2 + C3)} p1 * C1 - a1 * C1;$$

f2[C1_, C2_, C3_, CL_] := 
$$\frac{1}{1 + k (CL + C1 + C2 + C3)} p2 * C2 + a1 * C1 - a2 * C2;$$

f3[C1_, C2_, C3_, CL_] := a2 C2 - a3 C3;
f4[C1_, C2_, C3_, CL_] := 
$$\frac{pL * CL}{1 + k (C1 + C2 + C3)} * \left(1 - \frac{CL}{Lmax}\right) - gamma * CL;$$

```

Equilibrio y estabilidad

In[69]:=

```

Equilibrios = Solve[{f1[C1, C2, C3, CL] == 0 && f2[C1, C2, C3, CL] == 0 &&
  f3[C1, C2, C3, CL] == 0 && f4[C1, C2, C3, CL] == 0}, {C1, C2, C3, CL}];
J[C1_, C2_, C3_, CL_] := Evaluate[{
  {D[f1[C1, C2, C3, CL], C1], D[f1[C1, C2, C3, CL], C2],
   D[f1[C1, C2, C3, CL], C3], D[f1[C1, C2, C3, CL], CL]},

  {D[f2[C1, C2, C3, CL], C1], D[f2[C1, C2, C3, CL], C2],
   D[f2[C1, C2, C3, CL], C3], D[f2[C1, C2, C3, CL], CL]},

  {D[f3[C1, C2, C3, CL], C1], D[f3[C1, C2, C3, CL], C2],
   D[f3[C1, C2, C3, CL], C3], D[f3[C1, C2, C3, CL], CL]},

  {D[f4[C1, C2, C3, CL], C1], D[f4[C1, C2, C3, CL], C2],
   D[f4[C1, C2, C3, CL], C3], D[f4[C1, C2, C3, CL], CL]}
}];

Definition[J];

For[i = 1, i ≤ Length[Equilibrios], i++,
  peq = Equilibrios[[i]];
  J0 = J[peq[[1, 2]], peq[[2, 2]], peq[[3, 2]], peq[[4, 2]]];
  If[Det[J0] == 0, Print["En el punto de equilibrio",
    peq, "no puede aplicarse H-G porque el determinante es 0."],

    If[Re[Eigenvalues[J0][[1]]] < 0 && Re[Eigenvalues[J0][[2]]] < 0 &&
      Re[Eigenvalues[J0][[3]]] < 0 && Re[Eigenvalues[J0][[4]]] < 0,
      Print["La solución de equilibrio ", peq, " es estable."],
      Print["La solución de equilibrio ", peq, " es inestable."]]
  ]
]

```

La solución de equilibrio

$\{C1 \rightarrow 0., C2 \rightarrow -6.79913 \times 10^{11}, C3 \rightarrow -3.39957 \times 10^{11}, CL \rightarrow 1.04196 \times 10^{12}\}$ es inestable.

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 0., C3 \rightarrow 0., CL \rightarrow 0.\}$ es inestable.

La solución de equilibrio $\{C1 \rightarrow 0., C2 \rightarrow 0., C3 \rightarrow 0., CL \rightarrow 9.99585 \times 10^{11}\}$ es estable.

La solución de equilibrio

$\{C1 \rightarrow 0., C2 \rightarrow 1.47268 \times 10^{10}, C3 \rightarrow 7.36338 \times 10^9, CL \rightarrow 0.\}$ es inestable.

La solución de equilibrio $\{C1 \rightarrow -1.13837 \times 10^{11},$

$C2 \rightarrow -5.97644 \times 10^{11}, C3 \rightarrow -2.98822 \times 10^{11}, CL \rightarrow 1.04156 \times 10^{12}\}$ es inestable.

La solución de equilibrio

$\{C1 \rightarrow 3.52211 \times 10^9, C2 \rightarrow 1.84911 \times 10^{10}, C3 \rightarrow 9.24555 \times 10^9, CL \rightarrow 0.\}$ es inestable.

B. ESTABILIDAD Y SIMULACIONES DE LOS MODELOS

B.4 Simulaciones de modelos de médula con clon leucémico

Pruebas en Python introduciendo el compartimento L.

Simulaciones para modelos con clon leucémico

```
[1]: %matplotlib inline
import numpy as np
import pylab as p
import matplotlib.pyplot as plt
import scipy.integrate as sp
from scipy.integrate import solve_ivp
import math
```

Parámetros

```
[2]: p1 = math.log(2);
p2 = math.log(2)/(1.5);
a1 = 0.168;
a2 = 0.144;
a3 = 0.288;
k = 10**(-10);

#Parámetros relacionados con el clon

Lmax=10**(12);
gamma=0.001*a3;
kL=k;    ## esto lo cambio en el caso en el que quiera diferenciar
#la intensidad de la señal de las leucémicas de las sanas

#Tomamos como valores iniciales el equilibrio en médula sana
#(Valores obtenidos en Mathematica)

#Iniciamos un clon L en 10
```

Modelo I Clon Leucémico

Señalización dada por el recuento de células de Transición

Caso 1: El clon se encuentra en el compartimento de las Pro-B $p_L = p_1$

```
[3]: pL=p1
C0=[7.1565*10**9,8.70833*10**10,2.6125*10**10,1]
```

```
[4]: def modelo1(C,t):
    c1=C[0]
    c2=C[1]
    c3=C[2]
    c4=C[3]

    dc1dt=p1*c1/(1+k*(c3)+kL*c4)-a1*c1
    dc2dt=p2*c2/(1+k*(c3)+kL*c4) +a1*c1-a2*c2
    dc3dt=a2*c2-a3*c3
    #En sL no influye L
    dc4dt=pL*(c4/(1+k*(c3)))*(1-c4/Lmax)-gamma*c4
    dCdt=[dc1dt,dc2dt,dc3dt,dc4dt]
    return dCdt
```

```
[5]: #en el intervalo (1,300), quiero 6000 puntos:

t=np.linspace(1,300,6000)

# almacenaje de las soluciones
c1 = np.empty_like(t)
c2 = np.empty_like(t)
c3 = np.empty_like(t)
c4 = np.empty_like(t)

# Condiciones iniciales
c1[0] = C0[0]
c2[0] = C0[1]
c3[0] = C0[2]
c4[0] = C0[3]
# Solución EDOs
#tomo desde 1 a 16000, ie, todos los puntos que he definido

for i in range(1,6000):

    tspan = [t[i-1],t[i]]

    C = sp.odeint(modelo1,C0,tspan)

    #Almacenamos los datos para dibujarlos
    c1[i] = C[1][0]
    c2[i] = C[1][1]
    c3[i] = C[1][2]
    c4[i] = C[1][3]
```

```

#Próxima condición inicial
C0 = C[1]

ax1 = plt.subplot(212)
#ax1.margins(0.05)          # Default margin is 0.05, value 0 means fit
ax1.plot(t, c1*100/(c1+c2+c3+c4), 'b-', label='c1(t)')
ax1.plot(t, c2*100/(c1+c2+c3+c4), 'r--', label='c2(t)')
ax1.plot(t, c3*100/(c1+c2+c3+c4), 'g', linestyle='dotted', label='c3(t)')
ax1.plot(t, c4*100/(c1+c2+c3+c4), 'purple', label='L(t)')
ax1.set_xlim([120,160])
ax1.set_ylim([0, 105])

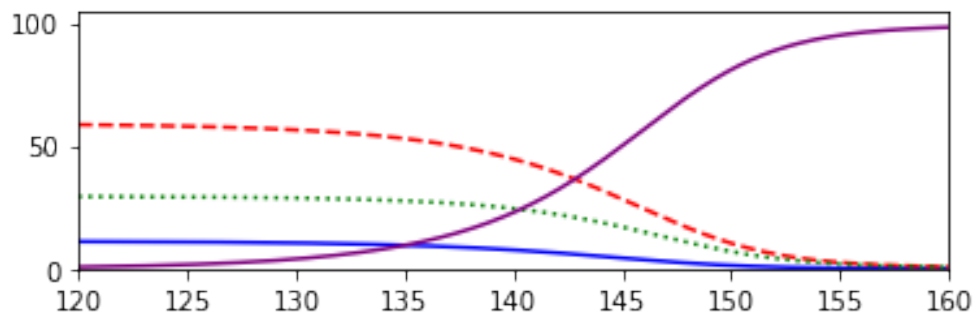
plt.show()

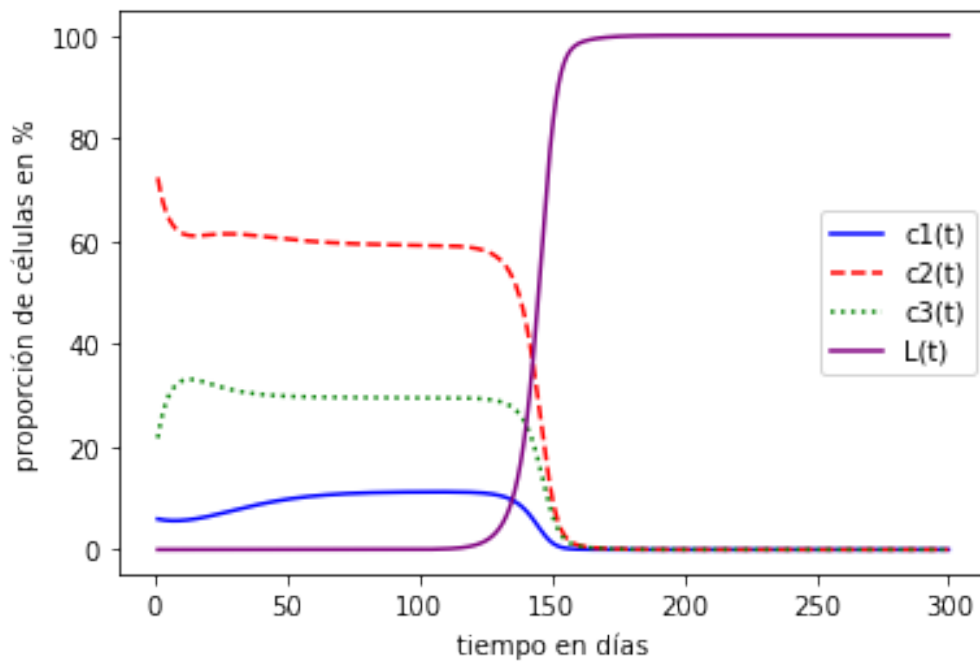
# Gráfica

plt.plot(t, c1*100/(c1+c2+c3+c4), 'b-', label='c1(t)')
plt.plot(t, c2*100/(c1+c2+c3+c4), 'r--', label='c2(t)')
plt.plot(t, c3*100/(c1+c2+c3+c4), 'g', linestyle='dotted', label='c3(t)')
plt.plot(t, c4*100/(c1+c2+c3+c4), 'purple', label='L(t)')
plt.ylabel('proporción de células en %')
plt.xlabel('tiempo en días')

plt.legend(loc='best')
mod1graf=plt.show()

```





```
[6]: C[1]
```

```
[6]: array([1.10338853e-01, 4.22187104e+01, 4.07764946e+01, 9.99584504e+11])
```

```
[ ]:
```

Caso 2: El clon se encuentra en el compartimento de las Pre-B $p_L = p_2$

```
[7]: pL=p2
C0=[7.1565*10**9,8.70833*10**10,2.6125*10**10,1]
```

```
[8]: def modelo1(C,t):
    c1=C[0]
    c2=C[1]
    c3=C[2]
    c4=C[3]

    dc1dt=p1*c1/(1+k*(c3)+kL*c4)-a1*c1
    dc2dt=p2*c2/(1+k*(c3)+kL*c4) +a1*c1-a2*c2
    dc3dt=a2*c2-a3*c3
    #En sL no influye L
    dc4dt=pL*(c4/(1+k*(c3)))*(1-c4/Lmax)-gamma*c4
    dCdt=[dc1dt,dc2dt,dc3dt,dc4dt]
    return dCdt
```

[9]: *#en el intervalo (1,300), quiero 6000 puntos:*

```
t=np.linspace(1,300,6000)

# almacenaje de las soluciones
c1 = np.empty_like(t)
c2 = np.empty_like(t)
c3 = np.empty_like(t)
c4 = np.empty_like(t)

# Condiciones iniciales
c1[0] = C0[0]
c2[0] = C0[1]
c3[0] = C0[2]
c4[0] = C0[3]
# Solución EDOs
#tomo desde 1 a 6000, ie, todos los puntos que he definido

for i in range(1,6000):

    tspan = [t[i-1],t[i]]

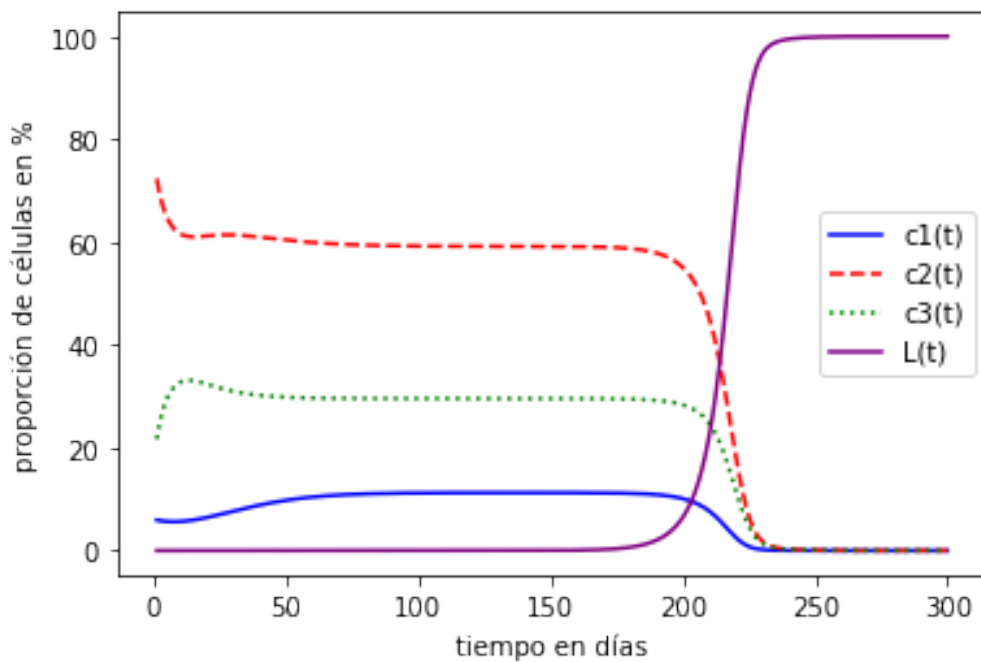
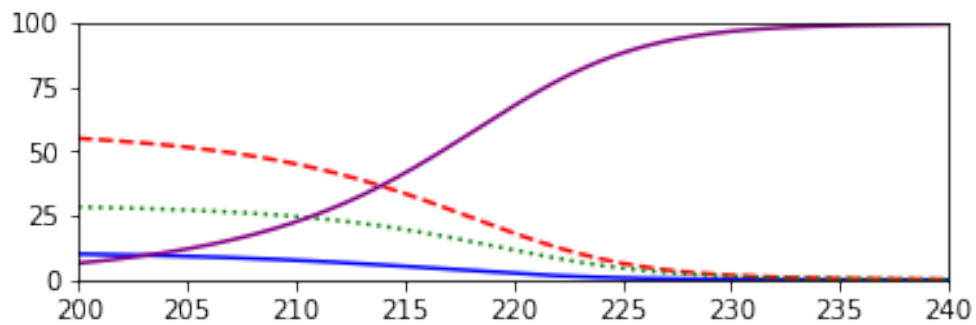
    C = sp.odeint(modelo1,C0,tspan)

    #Almacenamos los datos para dibujarlos
    c1[i] = C[1][0]
    c2[i] = C[1][1]
    c3[i] = C[1][2]
    c4[i] = C[1][3]
    #Próxima condición inicial
    C0 = C[1]

# Gráfica
ax1 = plt.subplot(212)
#ax1.margins(0.05) # Default margin is 0.05, value 0 means fit
ax1.plot(t, c1*100/(c1+c2+c3+c4), 'b-',label='c1(t)')
ax1.plot(t, c2*100/(c1+c2+c3+c4), 'r--',label='c2(t)')
ax1.plot(t, c3*100/(c1+c2+c3+c4), 'g',linestyle='dotted',label='c3(t)')
ax1.plot(t, c4*100/(c1+c2+c3+c4), 'purple',label='L(t)')
ax1.set_xlim([200,240])
ax1.set_ylim([0, 1*100])

plt.show()
```

```
plt.plot(t,c1*100/(c1+c2+c3+c4),'b-',label='c1(t)')
plt.plot(t,c2*100/(c1+c2+c3+c4),'r--',label='c2(t)')
plt.plot(t,c3*100/(c1+c2+c3+c4),'g',linestyle='dotted',label='c3(t)')
plt.plot(t,c4*100/(c1+c2+c3+c4),'purple',label='L(t)')
plt.ylabel('proporción de células en %')
plt.xlabel('tiempo en días')
plt.legend(loc='best')
modigraf=plt.show()
```



[10]: C[1]


```
[10]: array([1.16938818e+04, 8.56173071e+05, 8.14782797e+05, 9.99376684e+11])
```

Modelo II Clon Leucémico

Señalización dada por el recuento de todas las células

Caso 1: El clon se encuentra en el compartimento de las Pro-B $p_L = p_1$

```
[11]: pL=p1
      C0=[1.55331*10**9,1.89013*10**10,5.67039*10**9,1]

[12]: def modelo2(C,t):
      c1=C[0]
      c2=C[1]
      c3=C[2]
      c4=C[3]

      dc1dt=p1*c1/(1+k*(c1+c2+c3)+kL*c4)-a1*c1
      dc2dt=p2*c2/(1+k*(c1+c2+c3)+kL*c4) +a1*c1-a2*c2
      dc3dt=a2*c2-a3*c3
      #pl=p1 y en sL no influye L
      dc4dt=pL*(c4/(1+k*(c1+c2+c3)))*(1-c4/Lmax)-gamma*c4
      dCdt=[dc1dt,dc2dt,dc3dt,dc4dt]
      return dCdt

[13]: #en el intervalo (1,300), quiero 6000 puntos:

      t=np.linspace(1,300,6000)

      # almacenaje de las soluciones
      c1 = np.empty_like(t)
      c2 = np.empty_like(t)
      c3 = np.empty_like(t)
      c4 = np.empty_like(t)

      # Condiciones iniciales
      c1[0] = C0[0]
      c2[0] = C0[1]
      c3[0] = C0[2]
      c4[0] = C0[3]
      # Solución EDOs
      #tomo desde 1 a 6000, ie, todos los puntos que he definido

      for i in range(1,6000):

          tspan = [t[i-1],t[i]]
```

```

C = sp.odeint(modelo2,C0,tspan)

#Almacenamos los datos para dibujarlos
c1[i] = C[1][0]
c2[i] = C[1][1]
c3[i] = C[1][2]
c4[i] = C[1][3]
#Próxima condición inicial
C0 = C[1]

# Gráfica

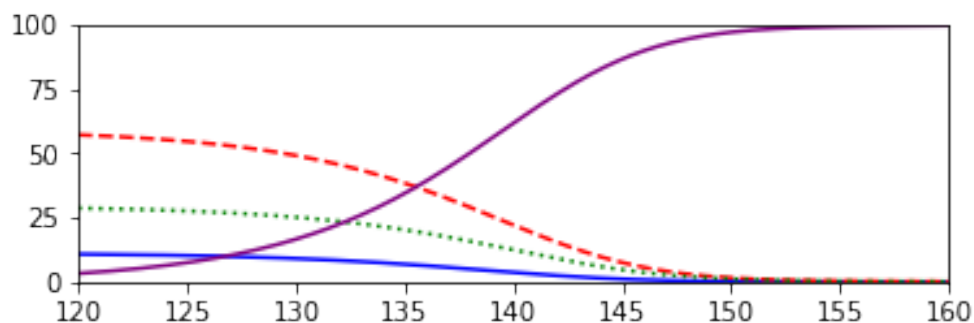
ax1 = plt.subplot(212)
#ax1.margins(0.05)           # Default margin is 0.05, value 0 means fit
ax1.plot(t, c1*100/(c1+c2+c3+c4), 'b-', label='c1(t)')
ax1.plot(t, c2*100/(c1+c2+c3+c4), 'r--', label='c2(t)')
ax1.plot(t, c3*100/(c1+c2+c3+c4), 'g', linestyle='dotted', label='c3(t)')
ax1.plot(t, c4*100/(c1+c2+c3+c4), 'purple', label='L(t)')
ax1.set_xlim([120,160])
ax1.set_ylim([0, 1*100])

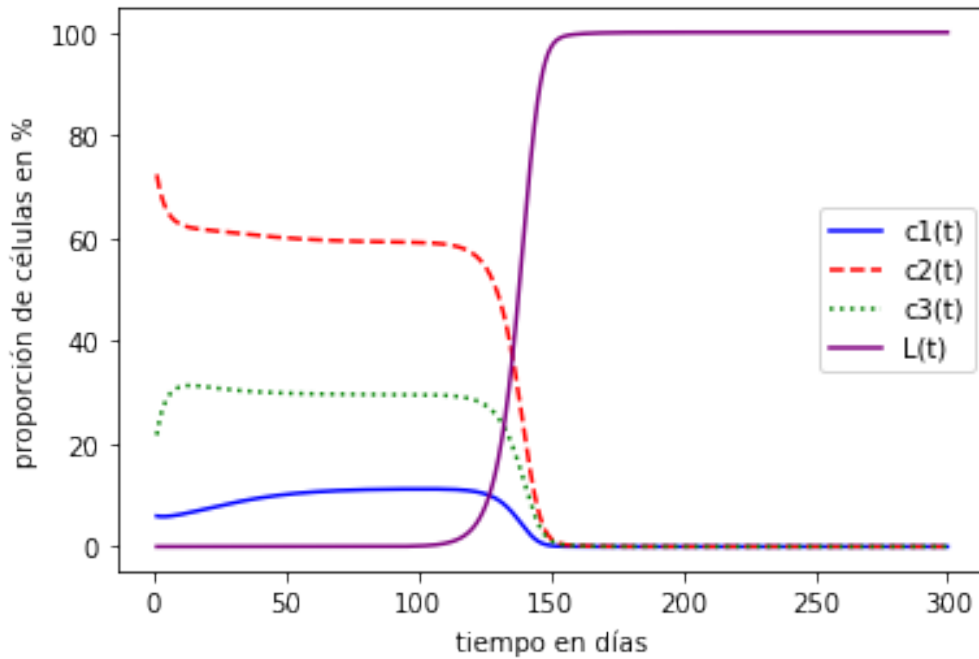
plt.show()

plt.plot(t,c1*100/(c1+c2+c3+c4), 'b-', label='c1(t)')
plt.plot(t,c2*100/(c1+c2+c3+c4), 'r--', label='c2(t)')
plt.plot(t,c3*100/(c1+c2+c3+c4), 'g', linestyle='dotted', label='c3(t)')
plt.plot(t,c4*100/(c1+c2+c3+c4), 'purple', label='L(t)')
plt.ylabel('proporción de células en %')
plt.xlabel('tiempo en días')

plt.legend(loc='best')
modigraf=plt.show()

```





```
[14]: C[1]
```

```
[14]: array([2.33859603e-02, 9.39019777e+00, 9.07090598e+00, 9.99584504e+11])
```

Caso 2: El clon se encuentra en el compartimento de las Pre-B $p_L = p_2$

```
[15]: pL=p2
C0=[1.55331*10**9,1.89013*10**10,5.67039*10**9,1]
```

```
[16]: def modelo2(C,t):
    c1=C[0]
    c2=C[1]
    c3=C[2]
    c4=C[3]

    dc1dt=p1*c1/(1+k*(c1+c2+c3)+kL*c4)-a1*c1
    dc2dt=p2*c2/(1+k*(c1+c2+c3)+kL*c4) +a1*c1-a2*c2
    dc3dt=a2*c2-a3*c3
    #pl=p1 y en sL no influye L
    dc4dt=pL*(c4/(1+k*(c1+c2+c3)))*(1-c4/Lmax)-gamma*c4
    dCdt=[dc1dt,dc2dt,dc3dt,dc4dt]
    return dCdt
```

[17]: *#en el intervalo (1,300), quiero 6000 puntos:*

```
t=np.linspace(1,300,6000)

# almacenaje de las soluciones
c1 = np.empty_like(t)
c2 = np.empty_like(t)
c3 = np.empty_like(t)
c4 = np.empty_like(t)

# Condiciones iniciales
c1[0] = C0[0]
c2[0] = C0[1]
c3[0] = C0[2]
c4[0] = C0[3]
# Solución EDOs
#tomo desde 1 a 6000, ie, todos los puntos que he definido

for i in range(1,6000):

    tspan = [t[i-1],t[i]]

    C = sp.odeint(modelo2,C0,tspan)

    #Almacenamos los datos para dibujarlos
    c1[i] = C[1][0]
    c2[i] = C[1][1]
    c3[i] = C[1][2]
    c4[i] = C[1][3]
    #Próxima condición inicial
    C0 = C[1]

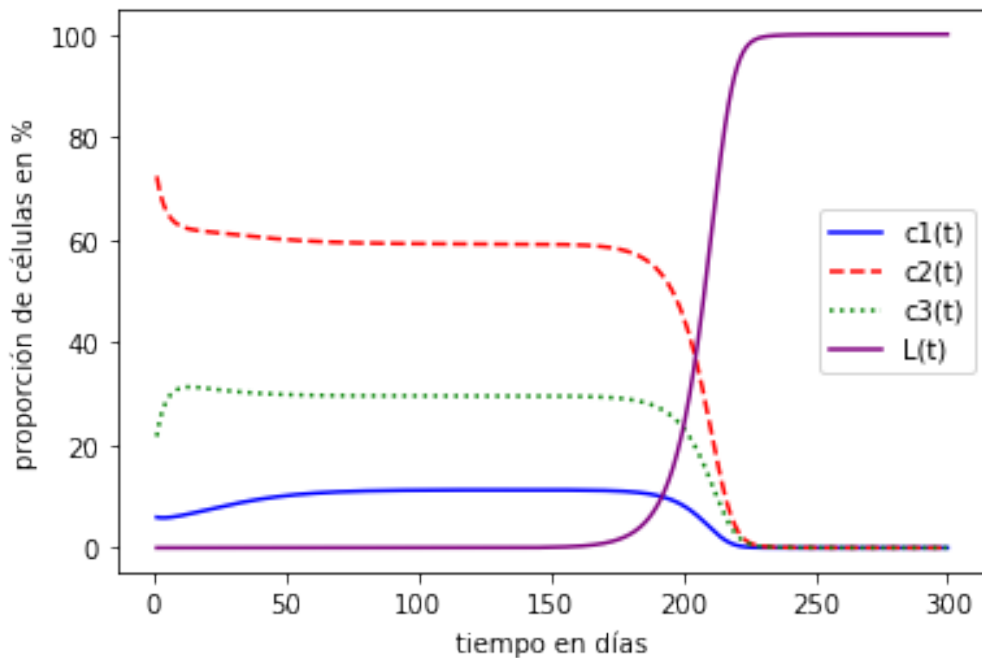
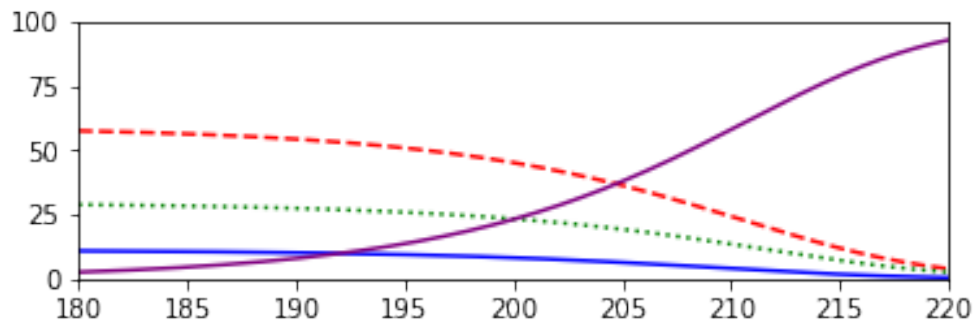
# Gráfica

ax1 = plt.subplot(212)
#ax1.margins(0.05) # Default margin is 0.05, value 0 means fit
ax1.plot(t, c1*100/(c1+c2+c3+c4), 'b-', label='c1(t)')
ax1.plot(t, c2*100/(c1+c2+c3+c4), 'r--', label='c2(t)')
ax1.plot(t, c3*100/(c1+c2+c3+c4), 'g', linestyle='dotted', label='c3(t)')
ax1.plot(t, c4*100/(c1+c2+c3+c4), 'purple', label='L(t)')
ax1.set_xlim([180,220])
ax1.set_ylim([0, 1*100])

plt.show()
```

```
plt.plot(t,c1*100/(c1+c2+c3+c4),'b-',label='c1(t)')
plt.plot(t,c2*100/(c1+c2+c3+c4),'r--',label='c2(t)')
plt.plot(t,c3*100/(c1+c2+c3+c4),'g',linestyle='dotted',label='c3(t)')
plt.plot(t,c4*100/(c1+c2+c3+c4),'purple',label='L(t)')
plt.ylabel('proporción de células en %')
plt.xlabel('tiempo en días')

plt.legend(loc='best')
mod1graf=plt.show()
```



ANEXO



Códigos

Se adjuntan los códigos implementados en el software *Python 3.7.3*. para la clasificación automática de los datos.

C.1 Algoritmo K-Means

Códigos usados para la representación del proceso de *Clustering* que sigue el algoritmo *K-Means* explicado en la Sección 5.3. Se adjunta a su vez, el código para el uso de la función *K-Means* para datos extraídos de un fichero de citometría de flujo, de la misma sección.

ALGORITMO K-MEANS

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Conjunto de puntos D

```
[2]: D = pd.DataFrame({
    'x': [13, 19, 30, 76, 27, 35, 22, 47, 43, 55, 52, 57, 55, 51, 58, 59, 67,
    ↪82, 75, 16],
    'y': [35, 40, 35, 48, 58, 42, 60, 83, 67, 72, 61, 65, 54, 87, 18, 6, 12, 9,
    ↪24, 76]
})
```

Etapas 0:

Designación aleatoria de centroides

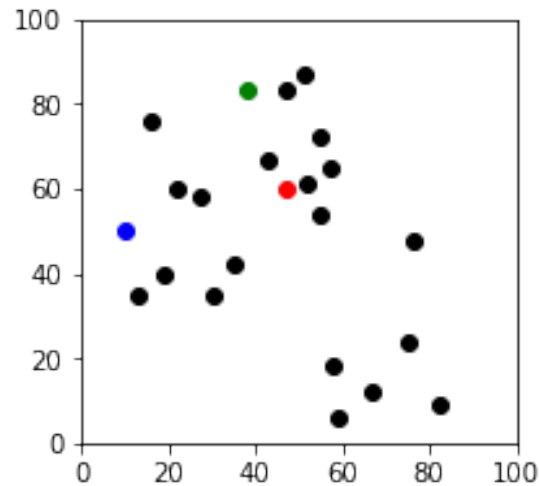
```
[9]: #semilla
np.random.seed(80)
#fijamos el número de clusters
k = 3
# centroides[i] = [c_i^x, c_i^y]=[x,y]
centroides = {
    i+1: [np.random.randint(0, 100), np.random.randint(0, 100)]
    for i in range(k)
}

fig = plt.figure(figsize=(3, 3))
plt.scatter(D['x'], D['y'], color='black')

#indicamos más de 3 colores para poder usar la función para otros casos
colores = {1: 'red', 2: 'green', 3: 'blue', 4: 'cyan', 5: 'magenta', 6: 'yellow',
    ↪7: 'orange', 8: 'purple', 9: 'fuchsia', 10: 'pink'}
for i in centroides.keys():
    plt.scatter(*centroides[i], color=colores[i])
plt.xlim(0, 100)
```



```
plt.ylim(0, 100)
plt.show()
```



Asignación de clusters a cada punto

```
[10]: ## Función para asignación de clusters

def asignacion(D, centroides):
    for i in centroides.keys():
        # calculamos distancia euclídea entre los puntos y los centroides:  $\sqrt{(x1 - x2)^2 - (y1 - y2)^2}$ 
        D['Distancia_{}'.format(i)] = (
            np.sqrt(
                (D['x'] - centroides[i][0]) ** 2 + (D['y'] - centroides[i][1])
            )
        )
        # mostramos la distancia a cada centroide
    distancias_cols = ['Distancia_{}'.format(i) for i in centroides.keys()]
    D['cluster'] = D.loc[:, distancias_cols].idxmin(axis=1)
    D['cluster'] = D['cluster'].map(lambda x: int(x.lstrip('Distancia_')))
    D['color'] = D['cluster'].map(lambda x: colores[x])
    return D
```

```
[13]: #Función para colorear cada punto del color de su cluster

def dibujar(D, centroides):
    fig = plt.figure(figsize=(3, 3))
    plt.scatter(D['x'], D['y'], color=D['color'], alpha=0.5, edgecolor='black')
```

```

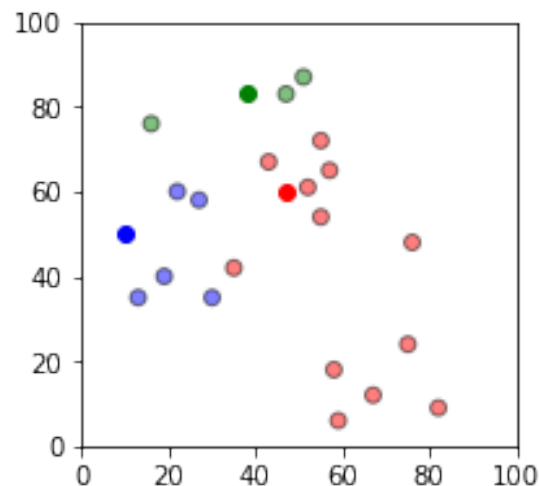
for i in centroides.keys():
    plt.scatter(*centroides[i], color=colores[i])
plt.xlim(0, 100)
plt.ylim(0, 100)
plt.show()

```

```

[16]: D = asignacion(D, centroides)
      dibujar(D,centroides)

```



```

[27]: #Veamos qué contiene D
      D.head(4)

```

```

[27]:    x  y  Distancia_1  Distancia_2  Distancia_3  cluster  color
0   13  35    58.587541    49.245717    15.114379         3   blue
1   19  40    54.502294    41.495670     7.310571         3   blue
2   30  35    42.432299    39.473290    11.493960         3   blue
3   76  48    29.231832    36.781662    51.753690         1    red

```

Etapa 1:

Reajuste de centroides

```

[18]: # Copiamos los últimos centroides antes de buscar los nuevos
import copy
centroidesAnteriores = copy.deepcopy(centroides)

def actualizar(k):
    for i in centroides.keys():
        centroides[i][0] = np.mean(D[D['cluster'] == i]['x'])
        centroides[i][1] = np.mean(D[D['cluster'] == i]['y'])

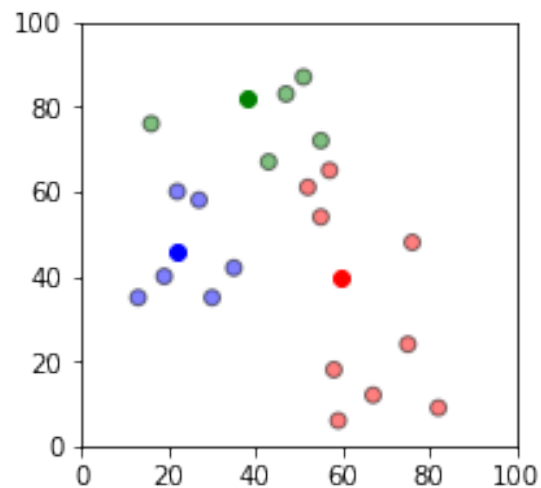
```

```
return k
```

```
[19]: centroidesAnteriores = copy.deepcopy(centroides)

centroides=actualizar(centroides)
D=asignacion(D,centroides)

dibujar(D,centroides)
```

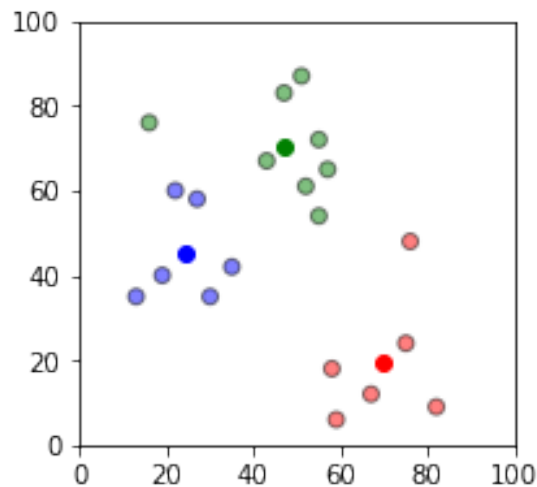


Repetición de etapas

```
[20]: #Lo paro en la iteración 30.
j=0
while centroidesAnteriores!=centroides and j<30:
    centroidesAnteriores = copy.deepcopy(centroides)
    centroides=actualizar(centroides)
    D=asignacion(D,centroides)
    j=j+1
print ("Proceso terminado. Ha realizado",j,"iteraciones")
```

Proceso terminado. Ha realizado 4 iteraciones

```
[21]: dibujar(D,centroides)
```



Definimos una función para diferenciar los clusters de manera visual

```
[22]: ## función que a cada x le hace corresponder un punto de la mediatriz del
      ↪ segmento (p1,p2)
def mediatrices(punto1,punto2, x):
    return (1/
    ↪ (2*(punto2[1]-punto1[1]))*(2*x*(punto1[0]-punto2[0])+(punto2[0])**2+punto2[1]**2-(punto1[1]
def mediatriz1(x):
    return mediatrices(centroides[1],centroides[2],x)
def mediatriz2(x):
    return mediatrices(centroides[2],centroides[3],x)
def mediatriz3(x):
    return mediatrices(centroides[3],centroides[1],x)

[25]: def colorearClusters(D,centroides):
    fig = plt.figure(figsize=(3, 3))
    plt.scatter(D['x'], D['y'], color=D['color'], alpha=0.5, edgecolor='black')
    for i in centroides.keys():
        plt.scatter(*centroides[i], color=colores[i])
    plt.xlim(0, 100)
    plt.ylim(0, 100)
    x=np.linspace(1,300,600)
    plt.fill_between(x, mediatriz1(x),0,
                     where=(mediatriz1(x)>mediatriz2(x)),
                     color='pink',
                     alpha=0.5) #proporción de color (transparencia)
    plt.fill_between(x, mediatriz2(x),0,
                     where=(mediatriz3(x)<mediatriz2(x)),
                     color='pink',
```

```

        alpha=0.5)

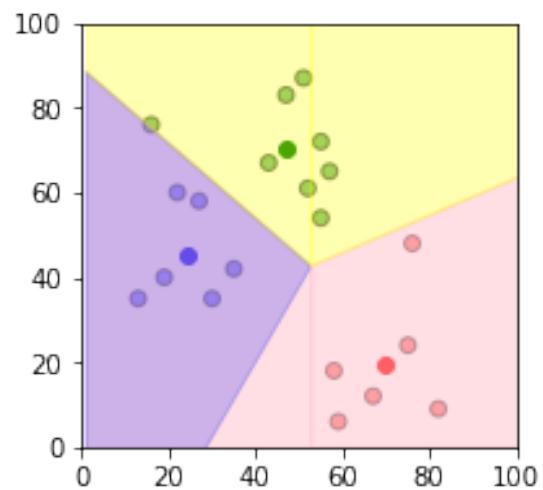
plt.fill_between(x, mediatriz2(x),mediatriz3(x),
                 where=(mediatriz3(x)<mediatriz2(x)),
                 color='blue',
                 alpha=0.2)

plt.fill_between(x, mediatriz2(x),100,
                 where=(mediatriz1(x)>mediatriz3(x)),
                 color='yellow',
                 alpha=0.3)
plt.fill_between(x, mediatriz1(x),100,
                 where=(mediatriz3(x)>mediatriz1(x)),
                 color='yellow',
                 alpha=0.3)

plt.show()

```

[26]: `colorearClusters(D,centroides)`



Lectura de fichero tipo .fcs y clusterización de los datos

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import cytoflow as flow
```

Función para leer un fichero .fcs

```
[2]: def lecturachange(datafile):
    tube=flow.Tube(file=datafile)
    namemetadata=flow.operations.import_op.autodetect_name_metadata(datafile)
    import_op = flow.ImportOp(tubes = [tube],name_metadata='$PnN')
    try:
        exp=import_op.apply()
    except KeyError:
        import_op = flow.ImportOp(tubes = [tube],name_metadata='$PnS')
        exp=import_op.apply()
    dicti=exp.metadata['fcs_metadata'][datafile]
    exportation=[];
    npar=dicti['$PAR'];
    dicti['lista']=[]
    for i in range(1,npar+1):
        if '$P'+str(i)+'S' in dicti:
            aux=dicti['$P'+str(i)+'S'];
            if aux.isdecimal() or (aux[0].isdigit() and aux.
↪endswith(('A','B','C'))):
                dicti['lista'].append(['CD'+aux,dicti['$P'+str(i)+'N']])
            else:
                dicti['lista'].append([aux,dicti['$P'+str(i)+'N']])
        else:
            dicti['lista'].append([dicti['$P'+str(i)+'N']])
    lista=dicti['lista']
    for j in range(0,len(lista)):
        markers=lista[j];
        if len(markers)>1 and markers[0] not in exp.channels:
            exp.metadata[markers[0]]=exp.metadata[markers[1]]
```

```

exp.data=exp.data.rename(columns={markers[1]:markers[0]})

del exp.metadata[markers[1]]
return exp

```

... Transformación de datos .fcs a .csv

```

[3]: exp=lecturachange('C:\\Users\\Ana\\Downloads\\PythonDoc\\P7r.fcs')
exp.data.to_csv('C:\\Users\\Ana\\Downloads\\PythonDoc\\P7r.csv')
experimento=pd.read_csv('C:\\Users\\Ana\\Downloads\\PythonDoc\\P7r.csv')

```

Dataframe que solo coja las columnas CD10 y CD45 del fichero .csv

```

[13]: D = pd.DataFrame({
        'x': experimento['CD10'],
        'y': experimento['CD45']
    })

```

Búsqueda de centroides con el algoritmo K-Means

```

[14]: from sklearn.datasets.samples_generator import make_blobs
from sklearn.cluster import KMeans

```

```

[15]: kmeans = KMeans(n_clusters=3, init='k-means++', max_iter=300, n_init=10,
    random_state=60)
pred_y = kmeans.fit_predict(D)

```

```

[16]: kmeans.cluster_centers_

```

```

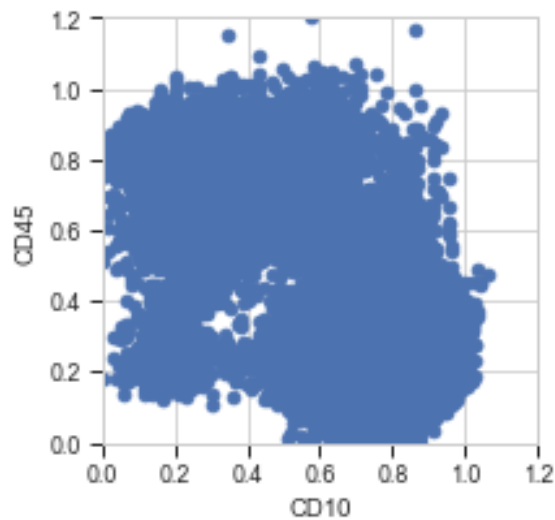
[16]: array([[0.76029712, 0.22610779],
            [0.30402961, 0.7312042 ],
            [0.74763437, 0.41178701]])

```

```

[17]: fig = plt.figure(figsize=(3, 3))
plt.scatter(D['x'], D['y'])
#plt.scatter(kmeans.cluster_centers_[ :, 0], kmeans.cluster_centers_[ :, 1],
    s=50, c='red')
plt.xlim(0, 1.2)
plt.ylim(0, 1.2)
plt.xlabel('CD10')
plt.ylabel('CD45')
plt.show()

```



... Representamos cada dato con el color que le corresponda según su cluster

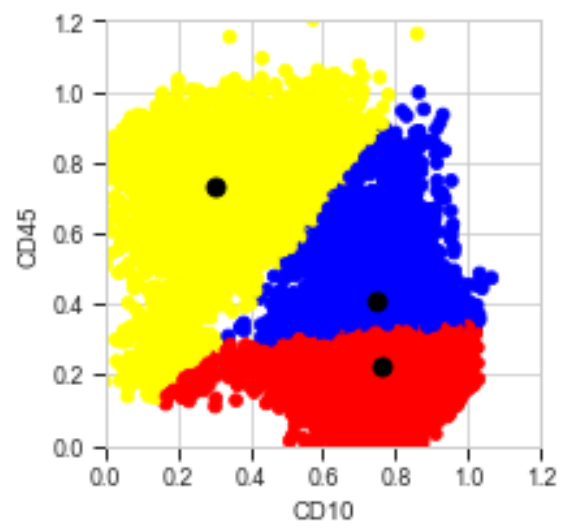
```
[18]: D['cluster'] = pred_y
```

```
[19]: data_df = pd.DataFrame(D)
data_df.columns=['x','y','cluster']
color_map= {0:'red',1:'yellow',2:'blue',3:'orange'}
data_df['true_color'] = data_df.cluster.map(color_map)
data_df.head(n=3)
```

```
[19]:
```

	x	y	cluster	true_color
0	0.867443	0.237520	0	red
1	0.807041	0.260285	0	red
2	0.710349	0.398987	2	blue

```
[20]: fig = plt.figure(figsize=(3, 3))
plt.scatter(x='x',y='y',c='true_color',data=data_df)
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s=50,
            c='black')
plt.xlabel("CD10")
plt.ylabel("CD45")
plt.xlim(0, 1.2)
plt.ylim(0, 1.2)
plt.show()
```

[]:

C. CÓDIGOS

C.2 Detección del clon

Programa con el que se cuentan las células de todos los compartimentos en cada paciente y verifica si se cumple o no las condiciones de médula sana como se indica en la Sección 5.5.

CONTADOR DE CÉLULAS

Función contador

[3]:

```
def contador(datafile,k):  
  
    listaCelulas=[]  
  
    ↪  
    ↪numCelulasCluster=len(datafile['Tipo_Celula'][datafile['Tipo_Celula']=='PRO-B'])  
    listaCelulas.append(numCelulasCluster)  
  
    ↪  
    ↪numCelulasCluster=len(datafile['Tipo_Celula'][datafile['Tipo_Celula']=='PRE-B'])  
    listaCelulas.append(numCelulasCluster)  
  
    ↪  
    ↪numCelulasCluster=len(datafile['Tipo_Celula'][datafile['Tipo_Celula']=='TRANS'])  
    listaCelulas.append(numCelulasCluster)  
  
    return listaCelulas
```

Ubicación de los archivos

```
[5]: ##CAMBIAR LA DIRECCIÓN!  
carpeta = 'C:\\Users\\Ana\\Downloads\\PythonDoc\\Datos R0 _ Ana Niño\\Datos R0_↪  
    ↪_ Ana Niño\\Sev+Jer\\'  
with os.scandir(carpeta) as ficheros:  
    ficheros = [fichero.name for fichero in ficheros if (fichero.is_file() and↪  
    ↪fichero.name.endswith('.fcs'))]  
    ficheros.sort() ##### esto ordena  
    #me aparacen todos los ficheros que acaban en .fcs
```

Bucle que toma los archivos .fcs del directorio miDir1 y los lee de miDir2 para trabajar con cada uno.

```
[6]: ##listas de almacenamiento

Pacientes=[] #identificador
TotalCelulas=[] #numero de células en la muestra
Pro=[] #proporción de células inmadura en tantos por 1
Pre=[] #PROPORCION
Trans=[] #PROPORCION
Centroides=[] #para la comprobación de los centroides en algún momento

n=len(ficheros);

for i in range(0,n):
    #Print para ver el progreso del bucle
    #print(ficheros[i].split('.')[0]+'': ('+str(i+1)+'/'+str(n)+''). '+str(np.
    ↳round((i+1)/len(ficheros),2)*100)+'%')
    #lo siguiente una vez q lo he hecho la primera vez,solo tendré que leer el
    ↳csv que importé una vez.

    miDir1=carpeta+ficheros[i] #de donde voy a leer el fichero

    miDir2='C:\\Users\\Ana\\Downloads\\PythonDoc\\DATOS\\'+ficheros[i] #donde
    ↳voy a guardar el csv

    exp=lecturachange(miDir1)
    exp.data.to_csv(miDir2)

    #necesitaría solo cambiar miDir2:
    experimento=pd.read_csv(miDir2)

    #que solo coja las columnas CD10 y CD45
    D = pd.DataFrame({
        'x': experimento['CD10'],
        'y': experimento['CD45']
    })

    kmeans = KMeans(n_clusters=3, init='k-means++', max_iter=300, n_init=10,
    ↳random_state=60)
    pred_y = kmeans.fit_predict(D)
    D['cluster'] = pred_y

    #AQUÍ VAMOS A DECIR A QUÉ TIPO CORRESPONDE CADA CLUSTER (No siempre los
    ↳centroides salen ordenados)
    centroide=[]
    centroide.append(kmeans.cluster_centers_[0])
```

```

centroide.append(kmeans.cluster_centers_[1])
centroide.append(kmeans.cluster_centers_[2])

tipocelula=['0','0','0']

for j in range(len(centroide)):

    if centroide[j][0]/centroide[j][1]==min(centroide[0][0]/
↪centroide[0][1],centroide[1][0]/centroide[1][1],centroide[2][0]/
↪centroide[2][1]):
        tipocelula[j]='TRANS'
    elif centroide[j][0]/centroide[j][1] == max(centroide[0][0]/
↪centroide[0][1],centroide[1][0]/centroide[1][1],centroide[2][0]/
↪centroide[2][1]):
        tipocelula[j]='PRO-B'
    else:
        tipocelula[j]='PRE-B'

celulas_map= {0:tipocelula[0],1:tipocelula[1],2:tipocelula[2]}

#una vez que hemos puesto un orden a los clusters (PreB, Transicion, ProB) ↪
↪o (ProB,PreB,Transición), asignamos su tipo a cada uno

D['Tipo_Celula'] = D.cluster.map(celulas_map)

#####

data_df = pd.DataFrame(D)
#data_df.columns=['x','y','cluster']
#data_df.head(n=4)
celulas=contador(data_df,3)

Pacientes.append(ficheros[i])

TotalCelulas.append(len(data_df))

Pro.append(celulas[0]/len(data_df))

Pre.append(celulas[1]/len(data_df))

Trans.append(celulas[2]/len(data_df))

```

[]:

MONTAMOS LA TABLA DE DATOS RECOGIDOS EN EL BUCLE

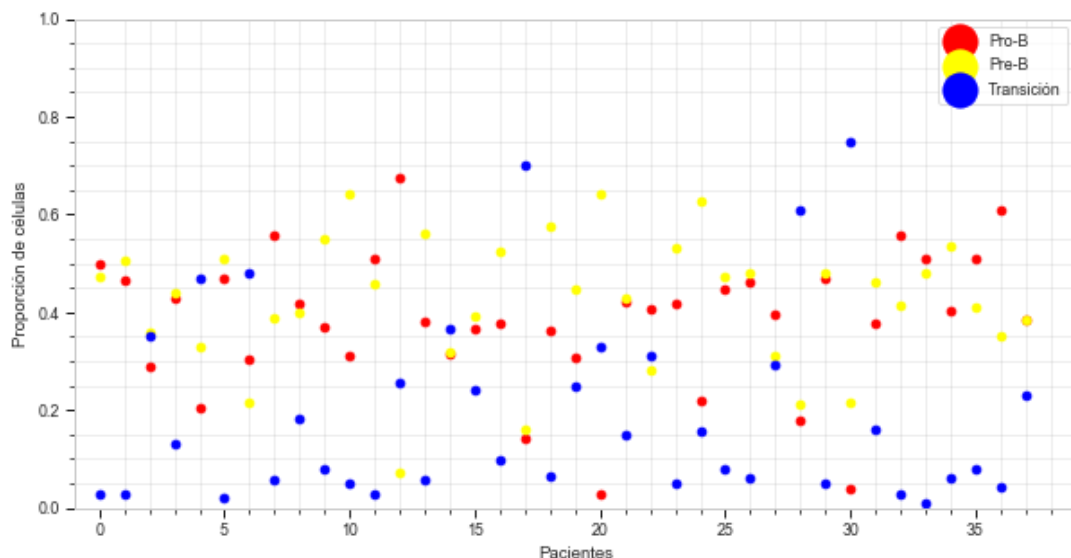
```
[7]: Data={'Pacientes':Pacientes,'Total_de_células':TotalCelulas, 'Pro-B':Pro,
↪ 'Pre-B':Pre,'Transición':Trans}
resumendatos = DataFrame(Data)
print(resumendatos)
```

	Pacientes	Total_de_células	Pro-B	Pre-B	Transición
0	P10r.fcs	156962	0.499006	0.473650	0.027344
1	P11r.fcs	162399	0.466025	0.507245	0.026730
2	P12r.fcs	161263	0.287140	0.360176	0.352685
3	P13.fcs	109575	0.427004	0.441405	0.131590
4	P13r.fcs	165335	0.204004	0.327686	0.468310
5	P14.fcs	162135	0.470855	0.507737	0.021408
6	P15.fcs	154432	0.304380	0.215363	0.480257
7	P17.fcs	93462	0.557521	0.386339	0.056140
8	P18.fcs	150673	0.419027	0.398014	0.182959
9	P19.fcs	143987	0.371165	0.550890	0.077945
10	P20.fcs	139709	0.310202	0.640617	0.049181
11	P21.fcs	161599	0.510900	0.459718	0.029381
12	P22.fcs	70144	0.674940	0.070099	0.254961
13	P23.fcs	124501	0.381788	0.562831	0.055381
14	P24.fcs	164261	0.312990	0.319157	0.367854
15	P25.fcs	26944	0.367280	0.393260	0.239460
16	P26.fcs	155757	0.376676	0.524708	0.098615
17	P28.fcs	30984	0.141170	0.158888	0.699942
18	P29.fcs	91699	0.362588	0.574815	0.062596
19	P31.fcs	161789	0.306257	0.445308	0.248435
20	P33.fcs	2249	0.025789	0.643842	0.330369
21	P34.fcs	169138	0.422265	0.428254	0.149481
22	P35.fcs	57712	0.407558	0.282385	0.310057
23	P36.fcs	158087	0.417896	0.533352	0.048752
24	P37.fcs	172241	0.218479	0.625525	0.155997
25	P40.fcs	136853	0.447312	0.471849	0.080839
26	P41.fcs	80105	0.460820	0.478435	0.060745
27	P42.fcs	162795	0.396787	0.310175	0.293037
28	P43.fcs	67708	0.177896	0.212634	0.609470
29	P45.fcs	148657	0.470075	0.481424	0.048501
30	P46.fcs	133226	0.037245	0.215716	0.747039
31	P47.fcs	153511	0.376970	0.463016	0.160015
32	P48.fcs	169807	0.557415	0.413681	0.028903
33	P49.fcs	162707	0.508349	0.480932	0.010719
34	P6r.fcs	137158	0.404344	0.536272	0.059384
35	P7r.fcs	85458	0.508215	0.411992	0.079794
36	P8r.fcs	147057	0.608043	0.351265	0.040692
37	P9r.fcs	171938	0.386000	0.384377	0.229623

```
[8]: DATOS=copy.copy(Data)
```

```
[9]: fig = plt.figure(figsize=(10, 5))
plt.scatter(x=range(len(Pacientes)),y='Pro-B',c='red',data=DATOS,label='Pro-B')
plt.
↳scatter(x=range(len(Pacientes)),y='Pre-B',c='yellow',data=DATOS,label='Pre-B')
plt.
↳scatter(x=range(len(Pacientes)),y='Transición',c='blue',data=DATOS,label='Transición')

#plt.scatter(kmeans.cluster_centers_[ :, 0], kmeans.cluster_centers_[ :, 1],
↳s=50, c='black')
plt.xlabel("Pacientes")
plt.ylabel("Proporción de células")
plt.xlim(-1, len(Pacientes)+1)
plt.ylim(0, 1)
plt.grid(False)
plt.minorticks_on()
plt.grid(b=True, which='minor', color='#999999', linestyle='--', alpha=0.2)
plt.legend(loc='best')
plt.show()
```



Detección del exceso de células

Una vez construida la tabla con todos los datos de las proporciones en las que se presenta cada compartimento, podemos estudiar cuál tiene una proporción ANORMAL de células, sabiendo que en médula sana, se presenta en mayor cantidad las Pre-B, luego las de Transición y por último las Pro-B.

```
[10]: for i in range(0,len(Pacientes)):
        #posiciones aux[0], aux[1], aux[2] corresponden con Pro, Pre y Trans
        #ORDEN EN MÉDULA SANA: PRE-TRANS-PRO
        #Si el máximo del vector no son las pre, habrá un fallo en el compartimento
        ↪máximo.

        ↪
        ↪aux=[resumendatos['Pro-B'][i],resumendatos['Pre-B'][i],resumendatos['Transición'][i]]
        posicionMax=aux.index(max(aux))

        if (aux[1]>aux[2]>aux[0]):
            print('El paciente',resumendatos['Pacientes'][i].replace('.
        ↪fcs',''),'tiene un exceso en el compartimento Pre-B')
        elif(posicionMax==0):
            print('El paciente',resumendatos['Pacientes'][i].replace('.
        ↪fcs',''),'tiene un exceso en el compartimento Pro-B')
        else:
            if(posicionMax==2):
                print('El paciente', resumendatos['Pacientes'][i].replace('.
        ↪fcs',''),'tiene un exceso en el compartimento de Transición')
                else: #maximo==PreB,y si no es el orden del principio, necesariamente
        ↪hay un fallo en pro-B
                print('El paciente', resumendatos['Pacientes'][i].replace('.
        ↪fcs',''),'tiene un exceso en el compartimento Pro-B')
```

El paciente P10r tiene un exceso en el compartimento Pro-B
 El paciente P11r tiene un exceso en el compartimento Pro-B
 El paciente P12r tiene un exceso en el compartimento Pre-B
 El paciente P13 tiene un exceso en el compartimento Pro-B
 El paciente P13r tiene un exceso en el compartimento de Transición
 El paciente P14 tiene un exceso en el compartimento Pro-B
 El paciente P15 tiene un exceso en el compartimento de Transición
 El paciente P17 tiene un exceso en el compartimento Pro-B
 El paciente P18 tiene un exceso en el compartimento Pro-B
 El paciente P19 tiene un exceso en el compartimento Pro-B
 El paciente P20 tiene un exceso en el compartimento Pro-B
 El paciente P21 tiene un exceso en el compartimento Pro-B
 El paciente P22 tiene un exceso en el compartimento Pro-B
 El paciente P23 tiene un exceso en el compartimento Pro-B
 El paciente P24 tiene un exceso en el compartimento de Transición
 El paciente P25 tiene un exceso en el compartimento Pro-B
 El paciente P26 tiene un exceso en el compartimento Pro-B
 El paciente P28 tiene un exceso en el compartimento de Transición
 El paciente P29 tiene un exceso en el compartimento Pro-B
 El paciente P31 tiene un exceso en el compartimento Pro-B
 El paciente P33 tiene un exceso en el compartimento Pre-B
 El paciente P34 tiene un exceso en el compartimento Pro-B

C.3 Clasificación de tipos de leucemia

Se adjunta el algoritmo usado para la clasificación de los pacientes según los marcadores dominantes en las células de sus muestras como se indica en la Sección 5.4, dando los resultados presentados en la Sección 5.5.

COMPARACIÓN DE LOS MARCADORES PARA OBTENER TIPOS LEUCEMIA

Bucle para leer todos los ficheros y clasificar según las proporciones de cada marcador

```
[7]: #n=len(ficheros)
n=38

for i in range(n):
    #Print para ver el progreso del bucle
    print(ficheros[i].split('.')[0]+'': ('+str(i+1)+'/'+str(n)+''). '+str(np.
    ↳round((i+1)/n,2)*100)+'%')

    miDir1=carpeta+ficheros[i] #de donde voy a leer el fichero

    miDir2='C:\\Users\\Ana\\Downloads\\PythonDoc\\ORFAO\\datos\\'+ficheros[i]
    ↳#donde voy a guardar el csv

    exp=lecturachange(miDir1)
    exp.data.to_csv(miDir2)
    experimento=pd.read_csv(miDir2)

    pro=0
    comun=0
    pre=0
    nocat=0
    numcelulas=len(experimento)

    for j in range(numcelulas):
        ↳
        ↳#celula=[experimento['CD10'][i],experimento['CD34'][i],experimento['CD38'][i],experimento['
        ↳
        ↳#celula=[experimento['CD10'][i],experimento['CD34'][i],experimento['CD38'][i]]

        if (celula[0]<1 and (celula[2]>1 and celula[1]>3)): pro=pro+1
        else:
```

```

        if(celula[0]>1.5) and (celula[2]>1.5 and celula[1]>1.5):
↪ comun=comun+1
        elif celula[1]<2: pre=pre+1
        else: comun=comun+1

    #print('Método 1: ')
    if (max(pro,comun,pre)==pro): print('El paciente ',ficheros[i].replace('.
↪ fcs',''),' tiene Leucemia PRO-B')
    elif (max(pro,comun,pre)==comun): print('El paciente ',ficheros[i].
↪ replace('.fcs',''),' tiene Leucemia COMÚN')
    else: print('El paciente ',ficheros[i].replace('.fcs',''),' tiene Leucemia_
↪ PRE-B')
    #print('las células no catalogadas han sido: ', nocat, 'en total')

```

P10r: (1/38). 3.0%
 El paciente P10r tiene Leucemia COMÚN
 P11r: (2/38). 5.0%
 El paciente P11r tiene Leucemia COMÚN
 P12r: (3/38). 8.0%
 El paciente P12r tiene Leucemia COMÚN
 P13: (4/38). 11.0%
 El paciente P13 tiene Leucemia PRE-B
 P13r: (5/38). 13.0%
 El paciente P13r tiene Leucemia PRE-B
 P14: (6/38). 16.0%
 El paciente P14 tiene Leucemia COMÚN
 P15: (7/38). 18.0%
 El paciente P15 tiene Leucemia PRE-B
 P17: (8/38). 21.0%
 El paciente P17 tiene Leucemia COMÚN
 P18: (9/38). 24.0%
 El paciente P18 tiene Leucemia COMÚN
 P19: (10/38). 26.0%
 El paciente P19 tiene Leucemia COMÚN
 P20: (11/38). 28.999999999999996%
 El paciente P20 tiene Leucemia COMÚN
 P21: (12/38). 32.0%
 El paciente P21 tiene Leucemia COMÚN
 P22: (13/38). 34.0%
 El paciente P22 tiene Leucemia PRE-B
 P23: (14/38). 37.0%
 El paciente P23 tiene Leucemia COMÚN
 P24: (15/38). 39.0%
 El paciente P24 tiene Leucemia COMÚN
 P25: (16/38). 42.0%
 El paciente P25 tiene Leucemia COMÚN
 P26: (17/38). 45.0%

El paciente P26 tiene Leucemia COMÚN
 P28: (18/38). 47.0%
 El paciente P28 tiene Leucemia PRE-B
 P29: (19/38). 50.0%
 El paciente P29 tiene Leucemia COMÚN
 P31: (20/38). 53.0%
 El paciente P31 tiene Leucemia PRE-B
 P33: (21/38). 55.00000000000001%
 El paciente P33 tiene Leucemia PRO-B
 P34: (22/38). 57.99999999999999%
 El paciente P34 tiene Leucemia COMÚN
 P35: (23/38). 61.0%
 El paciente P35 tiene Leucemia PRE-B
 P36: (24/38). 63.0%
 El paciente P36 tiene Leucemia PRE-B
 P37: (25/38). 66.0%
 El paciente P37 tiene Leucemia COMÚN
 P40: (26/38). 68.0%
 El paciente P40 tiene Leucemia COMÚN
 P41: (27/38). 71.0%
 El paciente P41 tiene Leucemia COMÚN
 P42: (28/38). 74.0%
 El paciente P42 tiene Leucemia COMÚN
 P43: (29/38). 76.0%
 El paciente P43 tiene Leucemia COMÚN
 P45: (30/38). 79.0%
 El paciente P45 tiene Leucemia COMÚN
 P46: (31/38). 82.0%
 El paciente P46 tiene Leucemia PRE-B
 P47: (32/38). 84.0%
 El paciente P47 tiene Leucemia COMÚN
 P48: (33/38). 87.0%
 El paciente P48 tiene Leucemia COMÚN
 P49: (34/38). 89.0%
 El paciente P49 tiene Leucemia COMÚN
 P6r: (35/38). 92.0%
 El paciente P6r tiene Leucemia COMÚN
 P7r: (36/38). 95.0%
 El paciente P7r tiene Leucemia COMÚN
 P8r: (37/38). 97.0%
 El paciente P8r tiene Leucemia PRE-B
 P9r: (38/38). 100.0%
 El paciente P9r tiene Leucemia COMÚN

[]:

Bibliografía

- [1] GLOBOCAN: Global Cancer Observatory, <https://gco.iarc.fr/>. 1
- [2] SEOM: Sociedad Española de Oncología Médica, <https://seom.org/>. 1, 5, 6
- [3] REDECAN: Red Española de Registros de Cáncer, <https://redecan.org/>. 1
- [4] AECC: Asociación Española Contra el Cáncer, <https://www.aecc.es/>. 2
- [5] Ching-Hon Pui, Leslie L. Robison, and A. Thomas Look. Acute lymphoblastic leukaemia. *The Lancet*, (9617), 2008. 2
- [6] Ching-Hon Pui, Dario Campana, and William E. Evans. Childhood acute lymphoblastic leukaemia—current status and future perspectives. *The lancet oncology*, 2(10):597–607, 2001. 2, 9
- [7] NCI: National Cancer Institute, <http://www.cancer.gov/>. 2, 6, 7, 8, 9, 10
- [8] American Cancer Society. Information and resources about Cancer: Breast, colon, lung, prostate, skin, <http://www.cancer.org>. 2, 5, 6, 7, 8, 9
- [9] Stuart H. Orkin and Leonard I. Zon. Hematopoiesis: an evolving paradigm for stem cell biology. *Cell*, 132(4):631–644, 2008. 3, 7, 8
- [10] Salvador Chulián, Alvaro Martínez-Rubio, Anna Marciniak-Czochra, Thomas Stiehl, Cristina Blázquez Goñi, Juan Francisco Rodríguez Gutiérrez, Manuel Ramirez Orellana, Ana Castillo Robleda, Víctor M. Pérez-García, and María Rosa. Dynamical properties of feedback signalling in b lymphopoiesis: A mathematical modelling approach. *arXiv preprint arXiv:2007.13526*, 2020. 3, 8, 9, 19, 22, 27, 28, 33, 35, 41, 49, 51, 59

BIBLIOGRAFÍA

- [11] James D. Murray. *Mathematical biology I: An introduction*, volume 17. Springer Science & Business Media, 2007. 3
- [12] Víctor M. Pérez-García, Susan Fitzpatrick, Luis A. Pérez-Romasanta, Milica Pesic, Philippe Schucht, Estanislao Arana, and Pilar Sánchez-Gómez. Applied mathematics and nonlinear sciences in the war on cancer. *Applied Mathematics and Nonlinear Sciences*, 1(2):423–436, 2016. 3
- [13] Philipp M. Altrock, Lin L. Liu, and Franziska Michor. The mathematics of cancer: integrating quantitative models. *Nature Reviews Cancer*, 15(12):730, 2015. 3
- [14] Real Academia Española. Diccionario de la lengua española (23.ª edición) Versión en línea, <http://dle.rae.es>, 2017. 5
- [15] Margaret Knowles and Peter J. Selby. *Introduction to the cellular and molecular biology of cancer*. Oxford University Press, 2005. 6
- [16] Douglas Hanahan and Robert A. Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674, 2011. 6
- [17] Miguel Ángel Medina Torres. Los signos distintivos del cáncer. *Cell*, 100:57–70, 2000. 6
- [18] Antonio Fasano and Adélia Sequeira. *Hemomath: The Mathematics of Blood*, volume 18. Springer, 2017. 7, 8
- [19] Terese Winslow. Terese Winslow LLC, Medical and Scientific Illustration, <https://www.teresewinslow.com/>. 7
- [20] Arkadiusz Pierzchalski, Anja Mittag, and Attila Tárnok. Introduction B: Recent advances in cytometry advances in applications. In *Methods in cell biology*, volume 103. Elsevier, 2011. 8
- [21] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular biology of the cell*. New York. Garland Science, 5th edition, 2002. 8
- [22] Fundación Josep Carreras contra la Leucemia, <https://www.fcarreras.org/es>. 8

- [23] Tommaso Lorenzi, Anna Marciniak-Czochra, and Thomas Stiehl. A structured population model of clonal selection in acute leukemias with multiple maturation stages. *Journal of mathematical biology*, 79(5):1587–1621, 2019. 9, 25, 45, 62
- [24] AEAL: Asociación Española de Afectados por Linfoma, Mieloma y Leucemia, <https://www.aeal.es/>. 9
- [25] Ellen G. Van Lochem, Vincent H. J. Van der Velden, Henk K. Wind, Jeroen G. Te Marvelde, Nomdo A. C. Westerdaal, and Jacques J. M. Van Dongen. Immunophenotypic differentiation patterns of normal hematopoiesis in human bone marrow: Reference patterns for age-related changes and disease-induced shifts. *Cytometry Part B: Clinical Cytometry*, 60(1):1–13, 2004. 9, 10
- [26] SEHOP: Sociedad Española de Hematología y Oncología Pediátrica. Programa para el estudio de la Terapéutica en Hemopatía Maligna (PETHEMA). Tratamiento de la Leucemia Aguda Linfoblástica de Nuevo Diagnóstico. 9
- [27] Álvaro Lassaletta Atienza. Leucemias. Leucemia linfoblástica aguda. *Pediatría Integral*, 20(6):380–389, 2016. 9
- [28] Claudia Mera Reina, Angélica Roa Lara, and Sandra Ramírez Clavijo. Hematopoietic stem cells, overview and pathways implied on their self-renewal mechanisms. *Ciencias de la Salud*, 5(1):67–89, 2007. 10
- [29] Kristina Anderson, Christoph Lutz, Frederik W. Van Delft, Caroline M. Bateman, Yanping Guo, Susan M. Colman, Helena Kempinski, Anthony V. Moorman, Ian Titley, John Swansbury, et al. Genetic variegation of clonal architecture and propagating cells in leukaemia. *Nature*, 469(7330):356–361, 2011. 10
- [30] Michael Heuser, Laura M. Sly, Bob Argiropoulos, Florian Kuchenbauer, Courteney Lai, Andrew Weng, Malina Leung, Grace Lin, Christy Brookes, Stephen Fung, et al. Modeling the functional heterogeneity of leukemia stem cells: role of STAT5 in leukemia stem cell self-renewal. *Blood, The Journal of the American Society of Hematology*, 114(19):3983–3993, 2009. 10
- [31] Juan Luis Romero Romero and Concepción García Vázquez. *Modelos y sistemas dinámicos*. Universidad de Cádiz, 1998. 11, 17

BIBLIOGRAFÍA

- [32] Martin Braun and Martin Golubitsky. *Differential equations and their applications*, volume 4. Springer-Verlag, 1993. 12
- [33] Anna Marciniak-Czochra, Thomas Stiehl, Anthony D. Ho, Willi Jäger, and Wolfgang Wagner. Modeling of asymmetric cell division in hematopoietic stem cells regulation of self-renewal is essential for efficient repopulation. *Stem cells and development*, 18(3):377–386, 2009. 19, 20, 23, 24, 59
- [34] Lourdes Barrera Ramírez, Drago Serrano, MA Elisa, Julia Pérez Ramos, Sainz Espuñes, Teresita Del Rosario, Ana Cecilia Zamora, Fabiola Gómez Arroyo, and Felipe Mendoza Pérez. Citometría de flujo: vínculo entre la investigación básica y la aplicación clínica. *Revista del Instituto Nacional de Enfermedades Respiratorias*, 17(1):42–55, 2004. 32
- [35] Jordi Petriz. Nuevas estrategias para la determinación de progenitores hematopoyéticos mediante citometría de flujo. *Medicina Clínica*, 117(6):218–221, 2001. 32
- [36] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011. 34
- [37] Nima Aghaeepour, Radina Nikolic, Holger H. Hoos, and Ryan R. Brinkman. Rapid cell population identification in flow cytometry data. *Cytometry Part A*, 79(1):6–13, 2011. 35
- [38] Robert F. Murphy. Automated identification of subpopulations in flow cytometric list mode data using cluster analysis. *Cytometry: The Journal of the International Society for Analytical Cytology*, 6(4):302–309, 1985. 35
- [39] Yongchao Ge and Stuart C. Sealfon. FlowPeaks: a fast unsupervised clustering for flow cytometry data via K-means and density peak finding. *Bioinformatics*, 28(15):2052–2058, 2012. 35
- [40] Francisco José Ortuño Giner and Alberto Orfao. Aplicación de la citometría de flujo al diagnóstico y seguimiento inmunofenotípico de las leucemias agudas. *Medicina Clínica*, 118(11):423–436, 2002. 39, 42

BIBLIOGRAFÍA

- [41] Dominique Bonnet and John E. Dick. Human acute myeloid leukemia is organized as a hierarchy that originates from a primitive hematopoietic cell. *Nature medicine*, 3(7):730–737, 1997. 45